

(12)特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関  
国際事務局(43) 国際公開日  
2004 年 4 月 22 日 (22.04.2004)

PCT

(10) 国際公開番号  
WO 2004/034395 A1

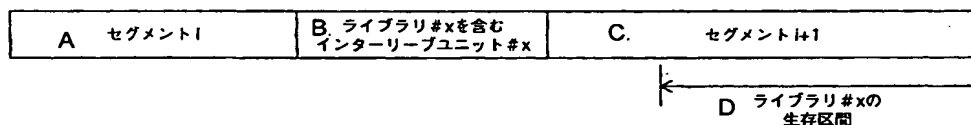
- (51) 国際特許分類: G11B 20/10, 27/00, G06F 3/06, H04N 5/92
- (21) 国際出願番号: PCT/JP2003/013026
- (22) 国際出願日: 2003 年 10 月 10 日 (10.10.2003)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ: 60/417,306 2002 年 10 月 10 日 (10.10.2002) US
- (71) 出願人 (米国を除く全ての指定国について): 松下電器産業株式会社 (MATSUSHITA ELECTRIC INDUSTRIAL CO., LTD.) [JP/JP]; 〒571-8501 大阪府 門真市 大字門真 1 0 0 6 番地 Osaka (JP).
- (72) 発明者: 池田 航 (IKEDA, Wataru) [JP/JP]; 〒534-0023 大阪府 大阪市都島区都島南通 2-1-3-1205 Osaka (JP). 岡田 智之 (OKADA, Tomoyuki) [JP/JP]; 〒631-0078 奈良県 奈良市富雄元町 1-8-19-303 Nara (JP). 上坂 靖 (UESAKA, Yasushi) [JP/JP]; 〒669-1348 兵庫県 三田市つつじが丘北 2-1-6-16 Hyogo (JP). 小塚 雅之 (KOZUKA, Masayuki) [JP/US]; 91007 カリフォルニア州 アーケディア、サウス ゴールデン ウェスト アベニュー 825、5号室 CA (US).
- (74) 代理人: 中島 司朗 (NAKAJIMA, Shiro); 〒531-0072 大阪府 大阪市北区豊崎三丁目 2 番 1 号 淀川 5 番館 6F Osaka (JP).
- (81) 指定国 (国内): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR,

[続葉有]

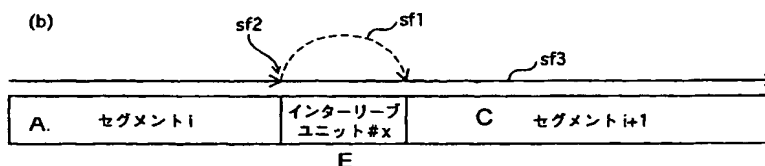
(54) Title: OPTICAL DISC, REPRODUCING DEVICE, PROGRAM, REPRODUCING METHOD, RECORDING METHOD

(54) 発明の名称: 光ディスク、再生装置、プログラム、再生方法、記録方法

(a)



(b)



A...SEGMENT i  
B...INTERLEAVE UNIT #x  
INCLUDING LIBRARY #x  
C...SEGMENT i+1  
D...EXISTENCE SECTION OF LIBRARY #x  
E...INTERLEAVE UNIT #x

(57) Abstract: An AV stream is divided into segments, and the segments are recorded on an optical disc. Before the segment to be reproduced *i*-th, an interleave unit is recorded. The interleave unit includes a library having the starting point of an existence section during the reproduction time of the *i*-th segment.

(57) 要約: AVストリームは、複数のセグメントに分割されて光ディスクに記録されており、複数セグメントのうち、*i*番目に再生されるものの前に、インターリーブユニットが記録されている。インターリーブユニットは、*i*番目のセグメントの再生時間内に、生存区間の開始点をもつライブラリを含む。



LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

- (84) 指定国 (広域): ARIPO 特許 (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア特許 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ特許 (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR),

OAPI 特許 (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:

- 国際調査報告書
- 請求の範囲の補正の期限前の公開であり、補正書受領の際には再公開される。

2 文字コード及び他の略語については、定期発行される各 PCT ガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

## 明 細 書

光ディスク、再生装置、プログラム、再生方法、記録方法

## 技術分野

- 5 本発明は、Blu-Ray Disc Read Only Memory(以下 BD-ROM と略す)等のデジタルストリーム用の光ディスク、光ディスクの再生装置、プログラム、再生方法、記録方法に関し、光ディスクにより映画作品等を頒布する技術に関する。

## 背景技術

- 10 光ディスクによる映画作品の頒布にあたっては、キャラクターについての映画作品と、キャラクターに関するゲームとを同じ光ディスクに記録して販売するというキャラクタービジネスがある。光ディスクに記録されるゲームは、クイズ形式の簡単なものから、三次元コンピュータ・グラフィックスを取り入れた本格的なものに  
15 発展する傾向がある。映画作品本編を構成する AV データと同じ光ディスクに、ゲームプログラムのような非 AV データを記録しておく先行技術には、以下の特許文献 1 に記載された技術がある。

＜特許文献 1＞ 特許 3069324 号公報

- しかしながらコンピュータ・グラフィックスを用いた本格的な  
20 ゲームは、動作のためにコンピュータ・グラフィックス描画のためのライブラリ等、様々なライブラリをメモリにロードしておき、アプリケーションプログラムに利用させる必要がある。そのため DVD プレーヤ等の民生機器のハードウェアスペックでは、かかる動作環境を満足させることはできず、パソコン等の他の動作環境  
25 での使用をユーザに喚起せねばならない。そうなると、ゲーム等の実行にあたっては、「推奨メモリ:128M バイト以上」というような動作環境での使用をユーザに求めることが多くなる。

- たとえ本格的なゲームが記録されていたとしても、ゲームの動作環境が、映画作品本編の動作環境と著しくかけ離れているのなら、ユーザの消費欲を冷めさせてしまう恐れがある。これでは、  
30

多大な開発費用をかけてゲームを制作し、映画作品と同じ光ディスクに記録する意義がなくなってしまう。

#### 発明の開示

本発明の目的は、ゲームを実現するようなプログラムを光ディスクに記録して頒布する場合に、そのプログラムの動作環境を、映画作品本編の動作環境に近づけることができる光ディスクを提供することである。

上記目的は、デジタルストリームを記録した光ディスクであって、デジタルストリームは、 $n$  個のセグメントに分割されて光ディスクに記録されていて、 $n$  個のセグメントのうち、第  $i$  番目に再生されるものの前に、インターリーブユニットが記録されており、ここで  $i, n$  は、 $i < n$  を満たす整数であり、インターリーブユニットは、第  $i$  番目のセグメントの再生に同期するプログラム、又は、第  $i$  番目のセグメントの再生に同期して表示されるデータを含む光ディスクにより達成される。

セグメント再生と同期実行すべきプログラムはインターリーブユニットという形でセグメントの前に配置されており、セグメントの再生時にあたっては光ピックアップによる読み取り位置をセグメントの先頭位置からずらせば、デジタルストリームを構成するセグメントを、同期実行すべきプログラムと共に読み出すことができる。同期処理に必要なプログラムは、セグメントの読み出し時にセグメントと併せて光ディスクから読み出せば足り、セグメントとの同期処理が済めば、メモリから削除すればよいので、デジタルストリーム再生の全区間を通じてプログラムを常駐しておく必要はない。デジタルストリームとの同期処理を行うべきプログラムが複数あっても、個々のプログラムの実行が終わる度に、プログラムからメモリを回収するという原則でリソース管理を行えば、セグメント再生との同期時に、プログラムをロードし得るだけのメモリを再生装置に実装しておけばよい。必要最低限のメモリ規模で、デジタルストリーム再生と、様々なプログラ

ムとの同期処理を実現することができる。

ここでインターリーブユニットは終了時点情報を含み、終了時点情報は、デジタルストリームの再生時間軸上におけるどの時点でメモリに読み出されたプログラム又はデータを削除するかを示していてもよい。

インターリーブユニットにおける終了時点情報の設定により、インターリーブユニットを何時からメモリに読み込み、何時までにインターリーブユニットをメモリから削除するかというリソース管理を、光ディスクのプレオーサリングの段階で決めておくことができる。これに基づくプログラミングを行えば、デジタルストリームにおける AV 再生との同期を行うようなゲームソフトを、小メモリ規模で実現することができる。ソフトハウスのプログラマにとって、AV 再生との同期を小メモリ規模で実現できる動作環境は魅力的であり、多くのソフトハウスの参入を促すことにより映画業界や民生機器業界を活性化することができる。

インターリーブユニットは開始時点情報を含み、開始時点情報は、インターリーブユニット内に含まれるプログラム又はデータの利用が、デジタルストリームの再生時間軸上におけるどの時点から、可能になるかを示していてもよい。

たとえば 3 つのライブラリが光ディスク上でバラバラに配置されたとしても、これらを同じ時点で利用可能とするよう、ヘッダにおける開始時点情報で調整することができる。

ここで第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの間には、前記インターリーブユニットの複製物が記録されていてもよい。

第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの境界は、プログラム又はデータの生存区間の途中にあたる位置であってもよい。

セグメントのうち、再生区間の開始点等頭出しが多く発生する位置の後にインターリーブユニット複製物を配置しておけば、

セグメント前に配置されているインターリーブユニットまでディスクシークを行わなくとも、アプリケーションプログラムによるライブラリ利用が可能となる。再生経路情報に基づく再生を行うため頭出しが多く発生したとしても、スムーズな読み取りが可能となる。

ここで第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの境界は、前記プログラムの生存区間の終了点より後にあたる位置であってもよい。

デジタルストリームの再生時間軸を未来から過去に遡って行き、プログラムの生存区間に到達した場合でも、不要なシークを行うことなく光ディスクからメモリへのプログラム読み出しを実現することができる。逆方向再生のような変則的な再生進行が行われる場合でも、メモリへのプログラム読み出しを実現出来るので、AV再生との同期を前提とした様々な処理を、逆方向再生時でも実現することができる。

ここで前記光ディスクにはプレイリスト情報、動的シナリオが記録されており、プレイリスト情報は、動画データにおける再生区間を示す情報を、再生順序に従って配列することにより再生経路を定義する情報であり、動的シナリオは、1つ以上の再生経路の再生手順を示すことにより、映像タイトルを定義する情報であり、インターリーブユニットは、識別情報を有しており、識別情報は、再生経路、再生区間、映像タイトル全体、映像タイトルの章の何れかを、プログラム又はデータの生存区間として示してもよい。

動的シナリオ、プレイリスト情報の番号等を用いて生存区間を定義しているので、インターリーブユニット同士の生存区間の重なる把握が容易となり、リソース管理を念頭においたインターリーブユニットの読み取り計画が立て易くなる。

#### 図面の簡単な説明

図1は、本発明に係る光ディスクの使用行為についての形態を

示す図である。

図 2 は、BD-ROM の構成を示す図である。

図 3 は、ディレクトリ構造を用いて BD-ROM の応用層フォーマットを表現した図である。

5 図 4 は、機能的な観点から、ファイルを分類した場合の分類図である。

図 5 は、BD-ROM が対象としているソフトウェアのレイアモデルを示す図である。

10 図 6 は、AV ストリームがどのように構成されているかを模式的に示す図である。

図 7 は、AV ストリームがどのように BD-ROM に記録されるかを模式的に示す図である。

図 8 は、ストリーム管理情報の内部構成を示す図である。

図 9 は、PL 情報の内部構成を示す図である。

15 図 10 は、PL 情報による間接参照を模式化した図である。

図 11 は、図 10 に示した PL とは、別の PL を定義する場合の一例を示す図である。

図 12 は、レイヤモデルの第 4 層 (動的シナリオ) における再生モードを示す図である。

20 図 13 は、Java 言語が対象とする制御ソフトウェアのレイアモデルを示す図である。

図 14 は、生存区間設定の対象となる AV ストリームを示す図である。

25 図 15 (a) は、BD-ROM に対しセグメントがどのように記録されるかを示す図である。

図 15 (b) は、セグメント  $i$ 、セグメント  $i+1$  を順次読み出す場合の、光ピックアップによる読み出し経路を示す。

図 16 は、セグメント  $i+1$  内に頭出しが発生した場合の光ピックアップの移動経路を示す図である。

30 図 17 は、AV ストリームの再生時間軸上の 4 つの時点

t1, t2, t3, t4 においてメモリへのライブラリの読み出し、及び、Java 仮想マシンのワーク領域へのライブラリロードがどのように行われるかを示す図である。

図 18 は、インターリーブユニットの構成を示す図である。

5 図 19 は、TMAP における ACCESS UNIT エントリーの改良を示す図である。

図 20 は、AV ストリームを構成するセグメントが読み取られる際のトラックバッファの状態遷移を示す図である。

10 図 21 は、映画作品の動画を背景画とし、コンピュータ・グラフィックスを前景画とした合成画面から構成されるゲームプログラムを示す図である。

図 22 (a) は、ライブラリ #1, #2, #3, #4 の生存区間を示す図である。

15 図 22 (b) は、各ライブラリの生存区間の IN 点と、セグメントとの関係を示す図である。

図 23 は、BD-ROM に対し、セグメントがどのように記録されているかを示す図である。

図 24 は、インターリーブユニットの内部構成を示す図である。

図 25 は、本発明に係る再生装置の内部構成を示す図である。

20 図 26 は、再生制御エンジン 12 の実行手順を示すフローチャートである。

図 27 は、図 25 に示した構成要素のうち、ライブラリに係るものを抜き出して描いた図である。

25 図 28 は、Java 言語が対象とするレイアモデルにおいて、表示用データが何処に位置するかを示す図である。

図 29 は、表示用データたるインターリーブユニットの構成を示す図である。

図 30 は、再生装置内部における表示用データが表示に至るまでの経路を示す。

30 図 31 は、セグメント内部のどこに、インターリーブユニット



複製物を組み込むかを示す図である。

図 3 2 は、セグメント i の内部に対して頭出しを行う場合、プログラムがどのように読み取られるかを示す図である。

5 図 3 3 は、図 2 2 に示した 4 つのライブラリ #1, #2, #3, #4 と、セグメントとの関係を示す図である。

図 3 4 (a) は、インターリーブユニット、セグメント、インターリーブユニット複製物の配置を示す図である。

図 3 4 (b) は、図 3 4 (a) に、i+1 番目のセグメントを書き加えた図である。

10 図 3 5 は、インターリーブユニット複製物がない場合に逆方向再生がどのように行われるかを示す図である。

図 3 6 は、インターリーブユニット複製物が配置された BD-ROM におけるシーク動作を示す図である。

15 図 3 7 は、図 2 2 に示した 4 つのライブラリ #1, #2, #3, #4 と、セグメントとの関係を示す図である。

図 3 8 は、ユーザにより逆方向再生操作が命じられた際の再生制御エンジン 1 2 の処理手順を示すフローチャートである。

図 3 9 は、第 5 実施形態に係る BD-ROM のファイル構成を示す図である。

20 図 4 0 は、PLMark 及び ClipMark の共通構成を示す図である。

図 4 1 (a) (b) は、プレイリスト #1 の再生中に出現する TimeEvent を定義する場合の PLMark の記述例を示す図である。

図 4 2 は、プレイリスト #1 の再生中において UserEvent を定義する場合の PLMark の記述例を示す図である。

25 図 4 3 は、ClipMark、PL にて TimeEvent、UserEvent が定義されている場合のインターリーブユニットの配置例を示す図である。

図 4 4 は、再生制御エンジン 1 2 の処理手順を示す図である。

30 図 4 5 は、インターリーブユニットに関する情報の間接参照がどのように行われるかを示す図である。

図 4 6 は、AV ストリームに組み込まれた 3 つのインターリーブユニットと、インターリーブユニット統合情報との対応を示す図である。

図 4 7 は、第 7 実施形態に係るインターリーブユニットにて、生存区間がどのように表記されるかを示す図である。

図 4 8 は、ロケータの記述によるプログラム又は表示用データの階層化配置を模式的に示す図である。

図 4 9 は、第 9 実施形態に係る再生装置の内部構成を示す図である。

図 5 0 は、第 1 1 実施形態に係る BD-ROM の製造工程を示すフローチャートである。

図 5 1 は、複数の表示用データを格納したインターリーブユニットを示す図である。

発明を実施するための最良の形態

(第 1 実施形態)

以降、本発明に係る光ディスクの実施形態について説明する。先ず始めに、本発明に係る光ディスクの実施行為のうち、使用行為についての形態を説明する。図 1 は、本発明に係る光ディスクの、使用行為についての形態を示す図である。図 1 において、本発明に係る光ディスクは、BD-ROM 1 0 0 である。この BD-ROM 1 0 0 は、再生装置 2 0 0、テレビ 3 0 0、リモコン 4 0 0 により形成されるホームシアターシステムに、映画作品を供給するという用途に供される。

続いて本発明に係る光ディスクの実施行為のうち、生産行為についての形態について説明する。本発明に係る光ディスクは、BD-ROM の応用層に対する改良により実施することができる。図 2 は、BD-ROM の構成を示す図である。

図 2 は、BD-ROM の構成を示す図である。本図の第 4 段目に BD-ROM を示し、第 3 段目に BD-ROM 上のトラックを示す。本図のトラックは、BD-ROM の内周から外周にかけて螺旋状に形成されて

いるトラックを、横方向に引き伸ばして描画している。このトラックは、リードイン領域と、ボリューム領域と、リードアウト領域とからなる。本図のボリューム領域は、物理層、ファイルシステム層、応用層というレイヤモデルをもつ。図2に示すようなデータフォーマットを、BD-ROMの応用層上に形成することにより本発明に係る光ディスクは、工業的に生産される。

図3は、ディレクトリ構造を用いて BD-ROM の応用層フォーマット(アプリケーション)を表現した図である。本図に示すように BD-ROM には、ROOT ディレクトリの下に BD-AV ディレクトリがあり、その下に JCLASS ディレクトリ、BROWSER ディレクトリがある。

BDAV ディレクトリの配下には、  
INFO.BD, XXX.M2TS, XXX.CLPI, YYY.PL, ZZZ.MOVIE といったファイルが存在する。JCLASS ディレクトリの配下には、ZZZ.CLASS というファイルが、BROWSER ディレクトリの配下には、ZZZ.HTM というファイルが配置されている。

図4は、機能的な観点から、これらのファイルを分類した場合の分類図である。本図において、第1層、第2層、第3層、第4層からなる階層が本図における分類を象徴的に示す。本図において XXX.M2TS は第2層に分類される。XXX.CLPI, YYY.PL は、第3層(静的シナリオ)に分類される。BDAV ディレクトリ配下の ZZZ.MOVIE、JCLASS ディレクトリ配下の ZZZ.CLASS、BROWSER ディレクトリ配下の ZZZ.HTM は、第4層に分類される。

本図の分類(第1層～第4層)は、図5に示すようなレイヤモデルを対象とした分類である。以降、図5を参照しながら、BD-ROM が対象としている、制御ソフトウェアのレイヤモデルについて説明する。

図5の第1層は、物理層であり、処理対象たるストリーム本体の供給制御である。この第1層に示すように、処理対象たるストリームは、BD-ROM だけではなく、HD、メモリカード、ネットワークといったあらゆる記録媒体、通信媒体を供給源としている。こ

れら HD、メモ리카ード、ネットワークといった供給源に対する制御(ディスクアクセス、カードアクセス、ネットワーク通信)が第1層の制御である。

第2層は、復号化方式のレイアである。第1層で供給されたストリームを、どのような復号化方式を用いて復号するのかを規定しているのがこの第2層である。本実施形態で採用する復号化方式は、MPEG2の復号化方式である。

第3層(静的シナリオ)は、ストリームの静的なシナリオを規定するレイアである。静的なシナリオとは、ディスク制作者によって予め規定された再生経路情報、ストリーム管理情報であり、これらに基づく再生制御を規定しているのがこの第3層(静的シナリオ)である。

第4層は、ストリームにおける動的なシナリオを実現するレイヤである。動的なシナリオとは、ユーザ操作や装置の状態によって再生進行を動的に変化させるためのシナリオであり、これらに基づく再生制御を規定しているのがこの第4層である。以降、このレイヤモデルに従い、ストリーム本体、静的なシナリオにあたるファイルについて説明してゆく。

先ず第2層に属するストリーム(XXX.M2TS)について説明する。

AVストリーム(XXX.M2TS)は、MPEG-TS(Transport Stream)形式のデジタルストリームであり、ビデオストリーム、1つ以上のオーディオストリーム、1つ以上の副映像ストリームを多重化することで得られる。ビデオストリームは映画の動画部分を、オーディオストリームは映画の音声部分を、副映像ストリームは、映画の字幕をそれぞれ示している。図6は、AVストリームがどのように構成されているかを模式的に示す図である。

AVストリームは(第4段目)、複数のビデオフレーム(ピクチャ p1, 2, 3)からなるビデオストリーム、複数のオーディオフレームからなるオーディオストリームを(第1段目)、PESパケット列に変換し(第2段目)、更にTSパケット列に変換し(第3段目)、同

じく副映像ストリーム(第7段目)を、PES パケット列に変換し(第6段目)、更に TS パケット列に変換して(第5段目)、これらを多重化することで構成される。この多重化は、オーディオフレームが、同じ時刻に BD-ROM から読み出されるべきビデオフレームの  
5 近くにくるように、ビデオフレームを格納した TS パケット、オーディオフレームを格納した TS パケットを並べるというものである。

かかる過程を経て生成された AV ストリームは、通常のコンピュータファイル同様、複数のセグメントに分割され、BD-ROM 上の  
10 領域に記録される。図7は AV ストリームがどのように BD-ROM に記録されるかを模式的に示す図である。

AV ストリームを構成する各セグメントが、どれだけの長さであり、BD-ROM においてどのアドレスに記録されているかは、ファイル管理情報 fkl に記述される。

15 ファイル管理情報 fkl には、AV ストリームを分割することで得られる3つのセグメント 1/3, 2/3, 3/3 のそれぞれについて、セグメントのアドレス(adrl, 2, 3)、長さ(length1, 2, 3)が記述されていることがわかる。AV ストリームは、1つ以上の ACCESS UNIT と  
20 となり、この ACCESS UNIT の単位で頭出し可能である。ACCESS UNIT とは、1つの GOP(Group Of Picture)と、この GOP と同時に読み出されるべきオーディオフレームとを含む最小デコード単位である。GOP は、過去方向および未来方向に再生されるべき画像との時間相関特性を用いて圧縮されている Bidirectionally  
25 predictive Predictive(B)ピクチャ、過去方向に再生されるべき画像との時間相関特性を用いて圧縮されている Predictive(P)ピクチャ、時間相関特性を用いず、一フレーム分の画像内での空間周波数特性を利用して圧縮されている Intra(I)ピクチャを含む。

尚、ファイル名 XXX.M2TS のファイルボディ「XXX」は、BD-ROM において AV ストリームに付与される3桁の識別番号を抽象化し  
30 ている。つまり本図における AV ストリームは、この XXX を用い

て一意に識別される。以上がストリーム(XXX.M2TS)についての説明である(ここでの 3 桁という桁数は例示に過ぎず、何桁でもよい。 )。

## 5      < 静的なシナリオ >

続いて、静的なシナリオであるファイル(XXX.CLPI,YYY.PL)について説明する。

ストリーム管理情報(XXX.CLPI)は、個々の AV ストリームについての管理情報である。図 8 は、ストリーム管理情報の内部構成を示す図である。AV ストリームはビデオストリーム、オーディオストリームを多重化することで得られ、AV ストリームは ACCESS UNIT と呼ばれる単位での頭出しが可能なので、各ビデオストリーム、オーディオストリームはどのような属性をもっているか、頭出し位置が AV ストリーム内の何処に存在するかが、ストリーム管理情報の管理項目になる。図中の引き出し線はストリーム管理情報の構成をクローズアップしている。引き出し線 hn1 に示すように、ストリーム管理情報(XXX.CLPI)は、ビデオストリーム、オーディオストリームについての「属性情報」と、ACCESS UNIT を頭出しするためのリファレンステーブルである「TMAP」とからなる。

属性情報(Attribute)は、破線の引き出し線 hn2 に示すようにビデオストリームについての属性情報(Video 属性情報)、属性情報数(Number)、AV ストリームに多重化される複数オーディオストリームのそれぞれについての属性情報(Audio 属性情報 #1 ~ #m)からなる。ビデオストリームについての管理情報は、破線の引き出し線 hn3 に示すようにそのビデオストリームがどのような圧縮方式で圧縮されたか(Coding)、ビデオストリームを構成する個々のピクチャデータの解像度がどれだけであるか(Resolution)、アスペクト比はどれだけであるか(Aspect)、フレームレートはどれだけであるか(Framerate)を示す。

一方、オーディオストリームについての属性情報(Audio 属性情報 #1~#m)は、破線の引き出し線 hn4 に示すようにそのオーディオストリームがどのような圧縮方式で圧縮されたか(Coding)、そのオーディオストリームのチャンネル番号が何であるか(Ch.)、何  
5 という言語に対応しているか(Lang)を示す。

タイムマップ(TMAP)は、複数の頭出し位置のアドレスを、時刻情報を用いて間接参照するためのリファレンステーブルであり、破線の引き出し線 hn5 に示すように複数のエントリー情報  
(ACCESS UNIT#1 エントリー情報、ACCESS UNIT#2 エントリー情報、  
10 ACCESS UNIT#3 エントリー情報……)と、エントリー情報数  
(Number)とからなる。各エントリー情報は、引き出し線 hn6 に示すように、対応する ACCESS UNIT の再生時間(Duration)と、対応する ACCESS UNIT のデータサイズ(Size)とを対応づけてなる。可  
15 変長符号圧縮方式が採用されるため、GOP を含む各 ACCESS UNIT  
のサイズや再生時間がバラバラであっても、この『エントリー情報』を参照することにより、任意の再生時刻から、その再生時刻  
に対応する ACCESS UNIT 内のピクチャデータへと頭出しを行うことが可能になる。尚、ファイル名「XXX.CLPI」のファイルボディ  
「XXX」は、ストリーム管理情報が対応している AV ストリームと  
20 同じ名称が使用される。つまり本図における AV ストリームのファイルボディは XXX であるから、AV ストリーム(XXX.M2TS)に対応していることを意味する。以上がストリーム管理情報についての説明である。続いてプレイリスト情報について説明する。

YYY.PL(プレイリスト情報)は、再生経路であるプレイリストを  
25 構成するテーブルであり、CellList からなる。図 9 は、PL 情報の内部構成を示す図である。

CellList は、複数の CELL 情報(CELL 情報 #1, #2, #3...#n)と、これら CELL 情報数(Number)とからなる。セル情報は、ポインタ情報であり、プレイリストを構成する 1 つ以上の論理的な再生区  
30 間を定義する。セル情報の構成は、引き出し線 hsl によりクロー

ズアップされている。この引き出し線に示すようにセル情報は、再生区間の In 点及び Out 点が属する AV ストリームの名称を示す『Stream Name』と、再生区間の始点を示す情報『IN 点情報』と、再生区間の終点を示す情報『Out 点情報』とから構成される。

- 5      セル情報の特徴は、その表記法にある。つまりタイムマップをリファレンステーブルとして用いた間接参照の形式で、再生区間が定義されている。図 10 は、PL 情報による間接参照を模式化した図である。本図において AV ストリームは、複数の ACCESS UNIT から構成されている。ストリーム管理情報内の TMAP は、これら
- 10   複数 ACCESS UNIT のセクタアドレスを、矢印 ay1, 2, 3, 4 に示すように指定している。図中の矢印 jy1, 2, 3, 4 は、CELL 情報による ACCESS UNIT の参照を模式化して示している。つまり、CELL 情報による参照(矢印 jy1, 2, 3, 4)は、TMAP を介することにより、AV ストリーム内に含まれる複数 ACCESS UNIT のアドレスを指定する
- 15   という間接参照であることがわかる。

- CELL 情報－ストリーム管理情報－AV ストリームの組みからなる BD-ROM 上の再生区間を『セル』という。PL 情報－ストリーム管理情報－AV ストリームの組みからなる BD-ROM 上の論理的な再生単位を『プレイリスト(PL と略す)』という。BD-ROM に記録さ
- 20   れた映画作品は、この論理的な再生単位(PL)にて構成される。論理的な再生単位にて、BD-ROM における映画作品は構成されるので、本編たる映画作品とは別に、あるキャラクタが登場するようなシーンのみを指定するような PL を定義すれば、そのキャラクタが登場するシーンのみからなる映画作品を簡単に制作することが
- 25   できる。図 11 は、図 10 に示した PL 情報(PL 情報#1)とは、別の PL(PL 情報#2)を定義する場合の一例を示す図である。

様々な PL 情報を定義するだけで、映画作品のバリエーションは増えるので、映画制作者の表現の幅を増やせることが、静的なシナリオの最大のメリットである。

- 30      また、BD-ROM における再生単位には、PL、CELL といったもの



の他、Chapter がある。Chapter は、1 つ、2 つ以上の CELL から構成される。

尚、PL 情報のファイル名におけるファイルボディ YYY は、BD-ROM において PL 情報に付与される 3 桁の識別番号を抽象化している。つまり本図における PL 情報は、この識別番号 YYY を用いて一意に識別される。PL 情報の識別番号を "YYY" と表現しているのは、PL 情報の識別番号が、AV ストリーム及び AV ストリーム管理情報の識別番号 XXX とは別の番号体系であることを意味している(ここでの 3 桁という桁数は例示に過ぎず、何桁でもよい)。

以上が、静的なシナリオについての説明である。続いて動的なシナリオについて説明する。

#### <動的なシナリオ>

続いて動的シナリオ「ZZZ.MOVIE」、「ZZZ.HTM」、「ZZZ.CLASS」について説明する。動的なシナリオは、AV ストリームの再生制御手順を示す。動的なシナリオによる再生制御手順は、装置に対するユーザ操作に応じて変化するものであり、プログラムの性質をもつ。ここでの動的な再生制御には、2 つのモードがある。2 つのモードのうち 1 つは、AV 機器特有の再生環境で、BD-ROM に記録された動画データを再生するモード(ノーマルモード)であり、もう 1 つは BD-ROM に記録された動画データの付加価値を高めるモード(エンハンスドモード)である。図 1 2 は、レイヤモデルの第 4 層における再生モードを示す図である。本図において第 4 層には、1 つのノーマルモードと、2 つのエンハンスドモードとが記述されている。1 つのノーマルモードは、DVD ライクな再生環境での再生モードであり MOVIE モードと呼ばれる。2 つのエンハンスドモードのうち、1 つ目は、Java 仮想マシンを主体とした再生モードであり、Java モードと呼ばれる。2 つ目のエンハンスドモードのうち、2 つ目はブラウザを主体とした再生モードであり、Browser モードと呼ばれる。

尚、ファイル名 ZZZ.MOVIE、ZZZ.CLASS、ZZZ.HTM におけるファ

イルボディ「ZZZ」は、BD-ROMにおいて動的シナリオに付与される3桁の識別番号を抽象化している。つまり本図におけるシナリオは、この識別番号ZZZを用いて一意に識別される。シナリオの識別番号を「ZZZ」と表現しているのは、シナリオの識別番号が、  
5 AVストリームの識別番号XXX、PL情報の識別番号YYYとは別の番号体系であることを意味している(ここでの3桁という桁数は例示に過ぎず、何桁でもよい。)

以降各モードの動的シナリオについてより詳しく説明する。

「ZZZ.MOVIE」は、MOVIEモードを対象とした動的シナリオである。

10 この動的シナリオでは、既存のDVD再生装置と良く似た再生制御を再生装置に実行させることができる。

「ZZZ.HTM」は、Browserモードを対象とした動的シナリオである。この動的シナリオでは、ネットワーク上のサイトをアクセスしたり、ファイルをダウンロードするような制御手順を記述する  
15 ことができる。

「ZZZ.CLASS」は、Javaモードを対象とした動的シナリオであり、Java言語のアプリケーションプログラムである。Java言語のアプリケーションプログラムなのでJavaモードの動的シナリオの実行主体は、Javaプラットフォームとなる。ここでJavaモード  
20 のアプリケーションと、Javaプラットフォームとの関係を図13を参照しながら詳しく説明する。図13は、Java言語が対象とするJavaプラットフォームのレイヤモデルを示す図である。Javaモードのアプリケーションは、このレイヤモデルの最上位のレイヤに位置する。このJavaモードのアプリケーションの下位に  
25 API(Application Interface)がある。更にその下位のレイヤにJavaプラットフォームがある。ネイティブ描画系は、再生装置が本来具備しているイメージ描画機能であり、Javaプラットフォームと同一階層にあたる。

Javaプラットフォームは、「Java仮想マシン(JavaVM)」、「コン  
30 フィグレーション」、「プロファイル」、「オプション」からなる。

Java 仮想マシンは、Java 言語で記述された Java モードアプリケーションを、再生装置における CPU のネイティブコードに変換して、CPU に実行させる。コンフィグレーションは、再生装置における簡単な入出力を実現する。プロファイルは、再生装置における IP 通信や、画面描画を行う。

「オプション」は、様々なライブラリを含む。これらは Java プラットフォームからは供給され得ない様々な機能を Java モードのアプリケーションに供給するものである。具体的にいえば、再生装置におけるセキュリティ確保の処理や BD-ROM～Java アプリケーション間の入出力がこのライブラリで規定されることになる。

再生装置において Java 仮想マシン、コンフィグレーション、プロファイル及びネイティブ描画系は、再生装置に予めインプリメントされる。また図 1 2 に示した第 1 層～第 3 層の制御ソフトウェアも再生装置に予めインプリメントされる。しかしオプションは、BD から読み出させねばならない。何故なら、Java モードのアプリケーションが必要とするオプションは、様々なものがあり、これらの全てを再生装置に予めインプリメントしておくことは困難だからである。以上が Java プラットフォームについての説明である。

インターリーブユニット「ILUsss.CLASS」は、インターリーブ記録されたファイルであり、プログラムやライブラリを格納している。「ILUsss.CLASS」におけるプログラムは、アプリケーション固有の処理を記述したバイトコード列であり、アプリケーションの主体となる。ライブラリとは、部品化されたプログラムであり、様々なアプリケーションにより利用されうるバイトコード列である。

以降、ライブラリについて詳しく説明する。本実施形態におけるライブラリは、AV ストリーム (PL) の再生時間軸上に生存区間をもつ。生存区間とは、アプリケーションによるライブラリの利用

が可能となる AV ストリーム (PL) の再生時間軸上の一期間をいう。

アプリケーションによるライブラリ利用とは、アプリケーションからの関数呼出に従い、ライブラリに含まれる関数を Java 仮想マシンに実行させることである。Java 仮想マシンにライブラリ  
5 を実行させるには、大きな前提条件がある。その前提条件とは、ライブラリを Java 仮想マシン内のワーク領域(ヒープ領域)にロードしておくことである。そのため、ライブラリ利用にあたってアプリケーションプログラムは、再生装置内のキャッシュメモリから Java 仮想マシンのワーク領域へのロードを命じておく必要  
10 がある。ここで BD-ROM に記録されているファイルが、BD-ROM から再生装置内のキャッシュメモリに予め読み込まれていれば、上述したロード命令は正常終了することになる。一方ライブラリが再生装置内のキャッシュメモリに読み込まれていなければ、アプリケーションプログラムによるロード命令は例外終了(エラー終了)  
15 することになる。ロード命令の正常終了を期待するなら、ライブラリファイルは、再生装置内のキャッシュメモリに読み出ししておくことが望ましい。そうしておけば再生装置内メモリにおけるライブラリを Java 仮想マシン内のワーク領域に何時でもロードすることができるからである。

20 ここでライブラリの生存区間を定めるということは、ライブラリを再生装置内のキャッシュメモリに読み出ししておくことと等価であると考えることができる。しかし BD-ROM 再生装置のような民生機器のハードウェアスペックでは、メモリが小規模であり、利用され得るライブラリが複数であれば、各ライブラリについて  
25 の生存区間を必要最低限にしなければならない。

ここで生存区間をどのように決めるべきかについて説明する。図 1 4 は、生存区間設定の対象となる AV ストリームを示す図である。AV ストリームの再生と同期するようなゲームアプリのプログラミングにあたっては、特定の映像が、出現している等の理由  
30 でアプリケーションプログラムによるライブラリ利用が多く予

想される期間を生存区間に定めることが望ましい。様々なライブラリの実行を、AV ストリームの再生と同期させることができるからである。例えば図 14 の AV ストリームは、本来映画作品を構成するものであり、人物同士の会話のシーンや、街並みを撮影したようなシーン等様々なシーンが含まれている。このうち街並みを撮影したようなシーンは、ゲームアプリケーションがコンピュータ・グラフィックスを描画するにあたって、背景画に利用される可能性が極めて高い。このようにゲームアプリによる利用の可能性が高い AV ストリーム上の一期間をライブラリの生存区間に定め、この生存区間において、再生装置のメモリにライブラリを読み出せるようにしておく。そうするとアプリケーションプログラムがロードを命じた場合、確実にワーク領域にライブラリをロードすることができる。以上のことを考慮すれば、生存区間を決めるにあたっては、アプリケーションプログラムの背景画として使用したいシーン等、ライブラリを同期させたいシーンが AV ストリームの再生時間軸上において何時出現するかを、プレオーサリングの段階で明らかにしておくことが必要となる。そして、ライブラリ同士の生存区間重複がなるべく発生しないように、各ライブラリの生存区間を定めることも、生存区間決定にあたっての責務となる。以上がライブラリの生存区間についての説明である。

続いてライブラリの記録形態であるインターリーブ記録について説明する。インターリーブ記録とは、BD-ROM に記録されるべき客体を AV ストリームを構成する複数セグメントの合間に記録するというものである。図 7 でいえば、「他のファイル」にあたる箇所に客体を記録しようとするのがインターリーブ記録である。ここでインターリーブ記録される客体を”インターリーブユニット”という。インターリーブ記録される客体がライブラリである場合、インターリーブユニットたるライブラリはどのように記録されるかを以下に説明する。ライブラリがインターリーブユニットである場合、インターリーブユニットはライブラリの生存

区間の IN 点を含むセグメントの直前に記録される。図 15 (a) は、BD-ROM に対しセグメントがどのように記録されるかを示す図である。AV ストリームが複数のセグメントに分割されて、そのうちセグメント  $i+1$  があるライブラリ #x の生存区間 IN 点を含んでいる場合、そのライブラリ #x を含むインターリーブユニットはセグメント  $i+1$  の直前に記録される。このように生存区間 IN 点を含むセグメントの直前に、記録されるというのがインターリーブユニットの特徴である。

インターリーブユニット #x、セグメント  $i$  は、図 15 (a) のように記録されているため、セグメントの読み出しは図 15 (b) に示すように行われる。図 15 (b) は、セグメント  $i$ 、セグメント  $i+1$  を順次読み出す場合の、光ピックアップによる読み出し経路を示す。セグメント  $i$  を読み込んだ後、本来なら光ピックアップは破線の矢印 sf1 に示すようなシークを行い、セグメント  $i+1$  を読み出さねばならない。セグメント  $i+1$  の前にはインターリーブユニットが存在しており、このインターリーブユニットはセグメント  $i+1$  の再生期間中に生存区間をもつライブラリを含んでいる。sf2 は、先行するインターリーブユニット #x が存在する場合の光ピックアップ読み出し位置であり、sf3 は光ピックアップによるディスクスキャンを示す。このディスクスキャン sf3 によりインターリーブユニット #x、セグメント  $i+1$  は、セグメント  $i$  の読み出し後、シークを行うことなくそのまま連続的に読み取られることになる。

図 15 (b) は、1 つの AV ストリームを構成するセグメントを連続して読み取るものであった。続いて、ライブラリの生存区間を含むセグメント内へと頭出しが発生した場合、光ピックアップによる読み取りがどのように行われるかを図 16 を参照しながら説明する。

図 16 は、セグメント  $i+1$  内に頭出しが発生した場合の光ピックアップの移動経路を示す図である。図中の頭出し位置 aP1 に光

ピックアップを移動させる場合にあって、セグメント  $i+1$  に先行するインターリーブユニットの存否が判定される。(2)は、先行するインターリーブユニットが存在する場合の光ピックアップ読み出し位置であり、(3)は光ピックアップによるディスクスキャンを示す。セグメントに先行するインターリーブユニットが存在する場合、インターリーブユニット先頭からセグメントへの光ピックアップスキャンがなされてインターリーブユニット、セグメントが再生装置により読み出される。このように光ピックアップを移動させれば、光ピックアップによる一回のスキャン(3)にて、動画再生に必要な ACCESS UNIT と、この ACCESS UNIT に関連するライブラリとを一度に読み出すことができる。

インターリーブユニットという形態でライブラリが記録されるため、現在の再生時点が生存区間に入ったタイミングでライブラリは、メモリに読み出されることになる。そのため、生存区間内にアプリケーションプログラムがロード命令を発すれば、再生装置内のメモリから Java 仮想マシン内のワーク領域へのロードが確実に成功する。図 17 は、AV ストリームの再生時間軸上の 4 つの時点  $t_1, t_2, t_3, t_4$  においてメモリへのライブラリの読み出し、及び、Java 仮想マシンのワーク領域へのライブラリロードがどのように行われるかを示す図である。

時点  $t_1$  は、ライブラリの生存区間外の時点である。生存区間外であるため、キャッシュメモリにはライブラリが読み出されていない。そのため、矢印  $py_1$  に示すようにアプリケーションプログラムがロード命令を発したとしてもライブラリを Java 仮想マシンワーク領域にロードすることはできず、ロードは矢印  $py_2$  に示すように例外終了することになる。

時点  $t_2$  はライブラリの生存区間 IN 点である。生存区間 IN 点に到達したため、矢印  $py_3$  に示すようにライブラリはメモリに読み出される。

時点  $t_3$  は、ライブラリの生存区間内であって、アプリケーション

アプリケーションプログラムによりロードが命じられた時点である。メモリにはライブラリがロードされているので、矢印 py5 に示すようなアプリケーションプログラムによるロード命令に応じてライブラリが Java 仮想マシンのワーク領域内にロードされることになる  
5 (矢印 py6)。これによりアプリケーションプログラムは、ライブラリ内の関数を自由に呼び出すことができる。

時点 t4 は、ライブラリ生存区間である。生存区間 OUT 点に達したため、ライブラリは矢印 py7 に示すようにメモリから削除される。そのため以降アプリケーションプログラムからロードが命  
10 じられたとしても、時点 t0 同様ロード命令は例外終了することになる。

続いてインターリーブユニットの内部構成について説明する。図 18 は、インターリーブユニットの構成を示す図である。インターリーブユニットは、本図に示すようにヘッダと、本体部とからなる。本実施形態において本体部とはライブラリのことである。  
15

ヘッダは、本体部にあたるライブラリがメモリにおいて何処に存在するかを示す『ロケータ』、本体部にあたるライブラリの生存区間の IN 点となる時点を示す『生存区間 IN 点情報』、本体部にあたるライブラリの生存区間の OUT 点となる時点を示す『生存  
20 区間 OUT 点情報』を含む。ロケータは、再生装置上のメモリにおいて、ライブラリを識別するために設けられる。ライブラリは、生存区間に関する情報が記述されたヘッダが付与され、インターリーブユニットという形態で BD-ROM に記録されるのである。

『ロケータ』を設けることの技術的意義について以下に説明する。インターリーブユニットに含まれるライブラリを、キャッシュメモリ上において、Java ディレクトリの配下にあるファイル「J001.CLASS」として扱いたい場合、ロケータを「CM://Java/J001.CLASS」と記述する。このロケータにおける「CM://」とは、再生装置内部におけるインターリーブユニット  
30 の置き場所がキャッシュメモリであることを示す。一方再生装置



内部におけるインターリーブユニットの置き場所がハードディスクであるなら、ロケータを「HD://」と記述すればよい。一方「Java/J001.CLASS」は、インターリーブユニットの置き場所のファイルパスである。再生装置は、インターリーブユニットの読み込みの際、キャッシュメモリ上に Java ディレクトリと、ファイル J001.CLASS とをクリエイト又はオープンし、インターリーブユニット内のライブラリを Java ディレクトリ配下のファイル J001.CLASS に格納しておく。こうすることでアプリケーションプログラムは、キャッシュメモリから Java 仮想マシン内ワーク領域へのライブラリ転送を命じる際、ファイルパス「Java/J001.CLASS」を用いてキャッシュメモリ上のライブラリを指定することができる。

『生存区間 IN 点情報』をインターリーブユニットヘッダに設けておくことの技術的意義について説明する。インターリーブユニットに含まれる生存区間の開始点は、原則 AV ストリームの再生時間軸上において、そのライブラリがメモリ上に存在している時点となる。しかしヘッダにおける生存区間 IN 点情報は、このメモリ上に存在している時点とは別の時点を、記述することができる。これは以下のようなケースを想定しているためである。

Java 言語におけるプログラミング作法には、3つのライブラリのうち1つのみがメモリがある状態でのアプリケーションプログラムによるライブラリの利用を許さず、3つのライブラリがメモリ上で揃ったときのみアプリケーションプログラムによるライブラリの利用を許すというものがある。ところがかかる3つのライブラリが3つのインターリーブユニットとして BD-ROM 上のバラバラの位置に記録された場合、3つのライブラリがメモリに読み込まれる時点は、ライブラリ毎に異なってくる。この場合3つのうち、1つ、2つがメモリ上にある状態でのアプリケーションプログラムによるライブラリ利用を禁じる必要があるので、各インターリーブユニットのヘッダにおける生存区間 In 点情報を、メ

メモリ上で3つのライブラリが揃う時点に設定しておく。

そうすると、3つのライブラリのうち、1つ目、2つ目のものがメモリにロードされた時点ではライブラリは利用可能とならず、3つ目のライブラリがメモリにロードされた段階で初めて利用可能となる。たとえ3つのライブラリがBD-ROM上でバラバラに配置されたとしても、これらを同じ時点で利用可能とするよう、ヘッダにおける生存区間 In 点情報で調整することができる。

『生存区間 Out 点情報』をインターリーブユニットヘッダに設けておくことの技術的意義について説明する。上述したように、BD-ROM 再生装置のハードウェアスペックではメモリ規模は限られており、キャッシュメモリ上に様々なライブラリを格納しておこうとすると、ライブラリ同士の上書きが発生してハングアップを招く恐れがある。そのためには、生存区間が過ぎたライブラリは速やかにキャッシュメモリから削除することが必要になる。かかる削除を促すべく、生存区間のヘッダに『生存区間 Out 点情報』が記されている。以上がインターリーブユニットのヘッダについての説明である。

生存区間 In 点情報及び生存区間 Out 点情報の表記について説明する。ここでAVストリームの再生時間軸とは、AVストリームに多重化されるビデオストリーム、オーディオストリームを構成するビデオフレーム、オーディオフレームの個々の再生タイミング、ビデオフレームのデコードタイミングの基準となる時間軸である。かかる再生時間軸上の任意の位置は、AVストリーム内の先頭ビデオフレームの時間情報を基準として、これとの相対値により表現される。かかる再生時間軸で再生タイミング、デコードタイミングが規定されているので、オーディオフレームービデオフレーム間の同期再生が可能になる。生存区間の In 点、Out 点も、再生時間軸上の一点と考えられるから、生存区間はビデオストリームの先頭ビデオフレームの時間情報を基準とした相対値で表現される。故に上述した生存区間 In 点情報及び生存区間 Out 点

情報は、ビデオストリームの先頭ビデオフレームの時間情報を基準とした相対値を用いて生存区間の In 点、Out 点を指定している。

ACCESS UNIT の頭出し時にあたって、ACCESS UNIT-インターリーブユニットを一度に読み出すには以下の改良が必要となる。それらの改良を以下に説明する。セグメントのうち、インターリーブユニットの直後にあたる ACCESS UNIT については、TMAP における ACCESS UNIT のエントリーを図 19 に示すように改変せねばならない。図 19 は、TMAP における ACCESS UNIT エントリーの改良を示す図である。本図における ACCESS UNIT エントリーは、インターリーブユニットの直後にあたる ACCESS UNIT について示している。本図において特徴的なのは、セグメントの先頭に位置する ACCESS UNIT#10, #20, #30 のエントリーに、インターリーブユニットのファイル名のファイルボディ (ILU001, ILU002, ILU003) が格納されている点である。このインターリーブユニットは、このセグメントに先行するインターリーブユニットである。かかるインターリーブユニットのファイル名を、ACCESS UNIT エントリーに格納しておくことで、セグメント先頭の ACCESS UNIT と、インターリーブユニットとをまとめて読み込むとの処理を効率的に行うことができる。

尚、ILUsss.CLASS の「sss」は、BD-ROM においてインターリーブユニットに付与される 3 桁の識別番号を抽象化している。つまり本図におけるインターリーブユニットは、この sss を用いて一意に識別される。インターリーブユニットの識別番号を「sss」と表現しているのは、インターリーブユニットが、AV ストリーム及び AV ストリーム管理情報の識別番号 XXX、プレイリスト情報の識別番号 YYY、動的シナリオの識別番号 ZZZ とは別の番号体系であることを意味している(ここでの 3 桁という桁数は例示に過ぎず、何桁でもよい。 )。

続いて、各セグメントにどれだけ連続長をもたせるかについて説明する。AV ストリームを分割して配置するとはいえ、AV スト

リームの記録にあたっては、このセグメントが読み出されるバッ  
 ファ(トラックバッファという)がアンダーフローしない長さに  
 定めねばならない。図20の上段は、AVストリームを構成するセ  
 グメントの1つを示す図であり、BD-ROMの位置a1から位置a2  
 5 までを占めている。図20下段のグラフは、本セグメントが読み  
 取られる際のトラックバッファの状態遷移を示す。このグラフの  
 横軸は、時間軸であり、縦軸はトラックバッファの蓄積量を表す。  
 このグラフにおける状態遷移は、 $V_a - V_b$ という増加率での単調増  
 加と、 $-V_b$ という減少率での単調減少とからなる。この単調増加  
 10 は、時点 $t_1$ から $t_2$ までの期間であり、単調減少は、時点 $t_2$ か  
 ら $t_3$ までの期間である。ここで時点 $t_1$ は、セグメントのうち、  
 位置a1の読み込み時点を示し、時点 $t_2$ は、セグメントのうち、  
 位置a1の読み込み時点を示す。時点 $t_3$ は次のセグメントの先頭  
 位置a3の読み込み時点を示す。 $V_a$ はトラックバッファへの入力  
 15 速度であり、 $V_b$ はトラックバッファからの出力速度である。時点  
 $t_2$ におけるトラックバッファの蓄積量 $B(t_2)$ は、式1により現さ  
 れる。

$$B(t_2) = (V_a - V_b) \times (t_2 - t_1) \dots\dots (式1)$$

20

セグメントの読出先たるトラックバッファが、かかる状態遷移  
 を行う場合、アンダーフローを避けるには、式2に示すように $t_1$   
 から $t_2$ までの増加量 $B(t_2)$ が、 $t_2$ から $t_3$ までの減少量 $V_b(t_3 - t_2)$   
 を上回ればよい。式2を満たすように、セグメントの連続長を決  
 25 めれば、トラックバッファはアンダーフローすることはない。

$$B(t_2) \geq V_b \times (t_3 - t_2) \dots\dots (式2)$$

以上が「ILUsss.CLASS」に格納されたライブラリの説明である。  
 30 続いて「ILUsss.CLASS」に格納されるプログラムについて説明す

る。「ILUsss.CLASS」に格納されたプログラムが、「ZZZ.CLASS」のアプリケーションと異なるのは、「ZZZ.CLASS」は動的シナリオの再生制御を記述するという処理に限られ、生存区間をもっていない。これに対し「ILUsss.CLASS」におけるアプリケーションは、動的シナリオの再生制御に限らず、ゲーム、クライアント処理等多様な制御の記述が可能であり、生存区間を有している（無論、「ILUsss.CLASS」内のプログラムは、再生制御のためのコードをもっている）。「ILUsss.CLASS」プログラムは、他のアプリケーション（「ZZZ.CLASS」に格納されたアプリケーションを含む）から、呼び出され得る区間を生存区間としている。そのためプログラムを格納した「ILUsss.CLASS」は、その生存区間の In 点より前にインターリーブ記録される。これにより「ILUsss.CLASS」内のプログラムは、「ZZZ.CLASS」のアプリケーションから呼出されうる時点より前に、メモリ 14 に読み出され、メモリ 14 から Java 仮想マシン内のワーク領域に転送されることになる。

インターリーブ記録には、よく似た先行技術がある。これは多重化という技術である。AV ストリームへの多重化と比較すると、本実施形態に示したインターリーブ記録には、次のような利点がある。多重化では、動画ストリーム及び音声ストリームと共に、ライブラリを AV ストリームに埋め込んでおくため、非 AV データたるライブラリの帯域を予め確保しておかなければならない。つまり非 AV データたるライブラリを動画ストリーム、音声ストリームと共に読み出すには、本来、動画ストリームに割り当てるべき帯域を低く抑えて、ライブラリを読み出すための帯域を確保せねばならない。映画作品の場合、結果的に映像、音声の帯域が狭くなり、映像あるいは音声の品質が落ちる結果となる。また映像、音声の帯域を狭くしないように、常に最適なライブラリ用の帯域を確保する場合は、各国の言語ごとに AV ストリームも多重化し直す必要が出てくる。これは映画作品を作る際には非常に手間がかかる。

それに対し、ライブラリを BD-ROM にインターリーブ記録しようとする場合、ライブラリは別のファイルとして記録されるため、インターリーブユニットをセグメントの合間に挟んだとしても、ファイルとしては論理的につながったデータとして利用することができる。そのため、AV ストリームの生成のための多重化工程は、ライブラリ作成とは別の行程になり、オーサリング作業の負担を減らすことができる。

以降、図 2 1 ～ 図 2 4 を参照しながらインターリーブユニットの設定例について説明する。この設定例では、図 2 1 に示すようなゲームプログラムを作成することを前提にしている。図 2 1 のゲームプログラムは、映画作品の動画を背景画とし、コンピュータ・グラフィックスを前景画とした合成画面から構成される。前景画たるコンピュータ・グラフィックスは、航空機を現したものであり、ユーザの操作に応じて姿勢や方向舵を変える。

一方背景画たる動画は、上空から都市を撮影した映像である。これらを合成した画像がユーザに提示されるため、ユーザはリアルなフライトシュミレーションを体感することができる。一方、ソフトウェアの制御面から考えれば、動画を背景画にしているため、Java モードのアプリケーションは AV ストリーム再生と同期して、コンピュータ・グラフィックス描画を行う必要がある。

この説明で引用するライブラリは、図 2 2 (a) に示すような生存区間をもつ 4 つのライブラリ (#1, #2, #3, #4) であるものとする。これらは、航空機の形状データの投影像を作成するレンダリングライブラリ、航空機表面の模様を示すテクスチャパターンを投影像に貼りつけるテクスチャマッピングライブラリ、投影像に陰影を施すシェーディング処理ライブラリ等であるものとする。図 2 2 (a) は、ライブラリ #1, #2, #3, #4 の生存区間を示す図である。本図の横軸は、AV ストリームの再生時間軸であり、この時間軸にそって、各ライブラリの生存区間を描いている。ライブラリ #1, #2, #3, #4 のうち、ライブラリ #1 の生存区間 IN 点が最も早

く、以降ライブラリ#2,#3,#4の生存区間IN点が続く。一方OUT点についても、インターリーブユニット#1の生存区間OUT点が最も早く、以降インターリーブユニット#2,#3,#4のOUT点が続く。

動画を構成するAVストリームは、BDにおいて4つのセグメント(セグメント1/4、セグメント2/4、セグメント3/4、セグメント4/4)に分割されてBDに記録される。AVストリームの分割境界は、ライブラリの生存区間のIN点を基準にしている。図22(b)は、各ライブラリの生存区間のIN点と、セグメントとの関係を示す図である。セグメントの境界(破線)は、時間軸におけるライブラリ#1,#2,#3,#4の生存区間のIN点からやや手前に設定されている。つまりライブラリの生存区間のIN点を基準に、AVストリームは分割されているのである。

図23は、BDに対し、セグメントがどのように記録されているかを示す。各インターリーブユニットは、セグメント1/4の前にインターリーブユニット#1を配置し、セグメント1/4~セグメント2/4の間にインターリーブユニット#2を配置し、セグメント2/4~セグメント3/4の間にインターリーブユニット#3を配置するように記録されている。図24は、インターリーブユニットの内部構成を示す図である。各インターリーブユニットは、ヘッダと、本体部たるライブラリ(ライブラリ#1,#2,#3,#4)とからなり、ヘッダはライブラリ#1,#2,#3,#4の生存区間のIN点、OUT点を示している。

インターリーブユニット、セグメントが図23のように記録されることにより、各ライブラリの生存区間直前に、各ライブラリはBDからメモリ上に読み出されることになる。またインターリーブユニットのヘッダには、生存区間のOUT点が表示されているので、AVストリームの再生時点が生存区間のOUT点に達すれば、メモリにおける各ライブラリは削除されることになる。かかる制御が実現可能になるので、生存区間が重なる期間において、各ライブラリを常駐させておくだけのメモリ資源を再生装置側に設け

ばよい。各ライブラリの生存区間から、必要最低限のメモリ規模を導きだし、これを満たすような推奨メモリ規模を定めれば良いので、動作環境を BD 再生装置のハードウェアスペックに近づけることができる。以上が本発明に係る光ディスクの実施形態である。

続いて本発明に係る再生装置の実施形態について説明する。図 25 は、本発明に係る再生装置の内部構成を示す図である。本図に示すように、再生装置は、BD ドライブ 1、トラックバッファ 2、デマルチプレクサ 3、ビデオデコーダ 4、ピクチャプレーン 5、オーディオデコーダ 6、イメージプレーン 8、イメージデコーダ 9、加算器 10、静的シナリオメモリ 11、再生制御エンジン 12、プレーヤレジスタ 13、メモリ 14、スイッチャ 15、DVD ライクモジュール 16、Java モジュール 17、BROWSER モジュール 18、UO マネージャ 19、トラックバッファ 21 から構成される。

BD-ROM ドライブ 1 は、BD-ROM のローディング／イジェクトを行い、BD-ROM に対するアクセスを実行する。

トラックバッファ 2 は、FIFO メモリであり、BD-ROM から読み出された ACCESS UNIT が先入れ先出し式に格納される。このトラックバッファ 2 に対する入力速度を図中の  $V_a$ 、出力速度を図中の  $V_b$  とした場合の、トラックバッファ 2 のバッファ状態は図 20 のグラフに示すものとなる。

デマルチプレクサ 3 は、トラックバッファ 2 から ACCESS UNIT を取り出して多重分離を行い、GOP を構成するビデオフレームと、オーディオフレームとを得てビデオフレームをビデオデコーダ 4 に出力し、オーディオフレームをオーディオデコーダ 6 に出力する。副映像ストリームを構成する副映像ユニットはイメージデコーダ 9 に出力する。デマルチプレクサ 3 による多重分離は、TS パケットを PES パケットに変換するという変換処理を含む。

ビデオデコーダ 4 は、デマルチプレクサ 3 から出力されたビデ



オフフレームを復号して非圧縮形式のピクチャをピクチャプレーン 5 に書き込む。

ピクチャプレーン 5 は、非圧縮形式のピクチャを格納しておくためのメモリである。

- 5     オーディオデコーダ 6 は、デマルチプレクサ 3 から出力されたオーディオフレームを復号して、非圧縮形式のオーディオデータを出力する。

- 10     イメージプレーン 8 は、一画面分の領域をもったメモリであり、コンピュータ・グラフィックスや字幕等、表示されるべきイメージが配置される。

- 15     イメージデコーダ 9 は、図 13 に示したネイティブ描画系にあたる構成要素であり、デマルチプレクサ 3 から出力された副映像ストリームを伸長してイメージプレーン 8 に書き込む。この他、Java 仮想マシンからの指示に応じてコンピュータ・グラフィックスの描画を行い、イメージプレーン 8 に書き込む。

- 20     加算器 10 は、ピクチャプレーン 5 に格納された非圧縮形式のピクチャデータに、イメージプレーン 8 に展開されたイメージを合成して出力する。図 21 に示した合成画面(動画を表示した画面の前にコンピュータ・グラフィックス(航空機)が合成されるような画面)は、この加算器 10 が、イメージプレーン 8 内のイメージと、ピクチャプレーン 5 内のピクチャとを合成することで出力される。合成にあたって加算器 10 は、混合率  $\alpha$  の指示を受け付けて、この混合率  $\alpha$  に応じて、ピクチャプレーン 5、イメージプレーン 8 に格納されている画像の画素値を混合して表示する。
- 25     これによりコンピュータ・グラフィックスの背後に、動画データの再生画像が透けて見えるような合成画像を表示することができる。混合率  $\alpha$  は、動的シナリオが変更できるようにしてもよい。

- 30     静的シナリオメモリ 11 は、カレントの PL 情報やカレントのストリーム管理情報を格納しておくためのメモリである。カレント PL 情報とは、BD-ROM に記録されている複数 PL 情報のうち、現

在处理対象になっているものをいう。カレントストリーム管理情報とは、BD-ROMに記録されている複数ストリーム管理情報のうち、現在処理対象になっているものをいう。

再生制御エンジン12は、AV再生機能(1)、プレイリストの再生機能(2)、再生装置における状態取得/設定機能(3)といった諸機能を実行する。再生装置のAV再生機能とは、DVDプレーヤ、CDプレーヤから踏襲した機能群であり、再生開始(Play)、再生停止(Stop)、一時停止(Pause On)、一時停止の解除(Pause Off)、Still機能の解除(still off)、速度指定付きの早送り(Forward Play(speed))、速度指定付きの逆方向再生(Backward Play(speed))、音声切り換え(Audio Change)、副映像切り換え(Subtitle Change)、アングル切り換え(Ungle Change)といった処理をユーザからの操作に応じて実行することである。PL再生機能とは、このAV再生機能のうち、再生開始や再生停止をPL情報に従って行うことをいう。このPL再生機能の実行により、再生制御エンジン12はレイヤモデルの第3層(静的なシナリオに基づく再生制御)の役割を果たす。再生制御エンジン12は、AV再生機能をユーザからの操作に従って実行する。一方、機能(2)~(3)を、DVDライクモジュール16~BROWSERモジュール18からの関数呼出に応じて実行する。つまり再生制御エンジン12は、ユーザ操作による指示やレイヤモデルにおける上位層からの指示に応じて、自身の機能を実行する。

プレーヤレジスタ13は、32個のSystem Parameter Registerと、32個のGeneral Purppose Registerとからなる。これらのレジスタの格納値は変数SPRM, GPRMとしてプログラミングに利用される。System Parameter Register、及び、General Purppose Registerは、DVDライクモジュール16~BROWSERモジュール18から分離した再生制御エンジン12で管理されるため、たとえ再生モードの切り換わりが生じたとしても、切換後の再生モードを実行するモジュールは、再生制御エンジン12における

SPRM(0)～(31)、GPRM(0)～(31)を参照しさえすれば、再生装置の再生状態を知得することができる。

Player Status Register の設定値 (SPRM) がどのような意味をもつかは、以下に示す通りである。以下の SPRM(x) という表記は、x

5 番目の Player Status Register の設定値を意味する。

	SPRM(0)	:	Reserved
	SPRM(1)	:	デコード対象たるオーディオストリームのストリーム番号
10	SPRM(2)	:	デコード対象たる副映像ストリームのストリーム番号
	SPRM(3)	:	ユーザによるアングル設定を示す番号
	SPRM(4)	:	現在再生対象とされているタイトルの番号
15	SPRM(5)	:	現在再生対象とされている Chapter の番号
	SPRM(6)	:	現在再生対象とされている PL の番号
	SPRM(7)	:	現在再生対象とされている CELL の番号
	SPRM(8)	:	現在の再生時点を示す時刻情報
20	SPRM(9)	:	ナビゲーションタイマのカウンタ値
	SPRM(10)	:	現在選択状態にあるボタンの番号
	SPRM(11)～(12)	:	Reserved
	SPRM(13)	:	ユーザによるパレンタルレベルの設定
	SPRM(14)	:	再生装置の映像再生に関する設定
25	SPRM(15)	:	再生装置の音声再生に関する設定
	SPRM(16)	:	再生装置における音声設定を示す言語コード
	SPRM(17)	:	再生装置における字幕設定を示す言語コード
30	SPRM(18)	:	メニュー描画のための言語設定

SPRM(19)～(31) : Reserved

ここで SPRM(10)は、AV ストリームに属する各ピクチャデータが表示される度に更新される。つまり再生装置が新たなピクチャデータを表示させれば、その新たなピクチャデータの表示開始時刻 (Presentation Time)を示す値に SPRM(10)は更新される。この SPRM(10)を参照すれば、現在の再生時点を知得することができる。同じく SPRM(6)を参照すれば、現在再生対象とされている PL の番号を知得することができ、SPRM(7)を参照すれば、現在再生対象とされている CELL の番号を知得することができる。

メモリ 14 は、BD から読み出されたインターリーブユニットを格納しておくキャッシュメモリである。メモリ 14 がトラックバッファ 2 と異なるのは、メモリ 14 は FIFO (First In First Out) のキュー型のメモリではない点である。他の構成要素は、必要なデータをこのメモリ 14 から取り出すことができる。一方、他の構成要素からの要求に応じて、インターリーブユニットを削除する。

スイッチャ 15 は、BD ドライブ 1 の光ピックアップが BD-ROM 上のセクタから情報を読み取る度に、セクタからの読み取り情報をトラックバッファ 2、メモリ 14 の何れかに書き込む。読み取り情報をトラックバッファ 2、メモリ 14 の何れに書き込むかは、ファイル管理情報を参照してなされる。つまり、ファイル管理情報には、各ファイルを構成するセグメントの所在が、ファイル名と対応づけられて示されているので、光ピックアップによる現在の読取位置のセクタアドレスを、ファイル管理情報と照合することで、セクタからの読み取り情報は、トラックバッファ 2、メモリ 14 の何れかに書き込まれる。現在の読取位置を示すセクタアドレスがファイル XXX.M2TS の記録領域内のものなら、BD-ROM からの読み取り情報を順次トラックバッファ 2 に書き込んでゆく。

現在の読取位置を示すセクタアドレスがライブラリの記録領域内のものなら、BD-ROMからの読み取り情報を順次メモリ14に書き込んでゆく。BD-ROMからの読み取り情報はセクタ単位であるが、読み取り情報のトラックバッファ2、メモリ14への書き込みを繰り返せば1つのACCESS UNITやライブラリがトラックバッファ2、メモリ14上に得られることになる。

DVDライクモジュール16は、MOVIEモードの実行主体となるDVD仮想プレーヤであり、動的シナリオメモリ15に読み出されたカレントのMOVIEモードの動的シナリオを実行する。

Javaモジュール17は、図13に示したJavaプラットフォームにあたる構成要素であり、メモリ14に読み出されたライブラリを用いて、カレントのJavaモードの動的シナリオを実行する。

BROWSERモジュール18は、Browserモードの実行主体となるブラウザであり、動的シナリオメモリ15に読み出されたカレントのブラウザモードの動的シナリオを実行する。BROWSERモジュール18が使用することができるプロトコルには、HTTP、IP、ARP、RARP、TCP、telnet、SMTP、ftpがある。

UOマネージャ19は、リモコンや再生装置のフロントパネルに対してなされたユーザ操作を検出して、ユーザ操作を示す情報(以降UO(User Operation)という)を出力する。

動的シナリオメモリ21は、カレント動的シナリオを格納しておき、カレントの動的シナリオをDVDライクモジュール16～BROWSERモジュール18による処理に供するメモリである。カレント動的シナリオとは、BD-ROMに記録されている複数シナリオのうち、現在実行対象になっているものをいう。

以上がBDドライブ1～動的シナリオメモリ21の説明である。続いて本実施形態に係る再生装置がインターリーブユニット読み出しをどのように行うかについて説明する。インターリーブユニット読み出しは、再生制御エンジン12がPL再生を実行する

際、AV ストリームを構成する ACCESS UNIT と共にメモリ 14 に読み出される。PL 再生は、図 26 に示したフローチャートの処理手順に従ってなされる。再生制御エンジン 12 は、DVD ライクモジュール 16 ~ BROWSER モジュール 18 からの PlayPL 関数の呼出時に、図 26 のフローチャートに従い PL 再生を行う。PlayPL 関数とは PlayPL(XX, YY) という表記で呼び出される関数であり、第 1 引数 XX に指定される PL を、第 2 引数 YY に示される CELL から再生させるというものである。

図 26 のフローチャートにおいて処理対象たる PL を PLx、処理対象たる CELL を CELly、処理対象たる ACCESS UNIT を ACCESS UNITv とする。本フローチャートに先立ち、PLPlay 関数の引数で指定された PL を PLx に設定し、PLx をメモリに読み込んでから、処理対象たる CELL を特定する。関数呼出にもし引数指定がなければ、CELly を PLx の先頭 CELL に設定し、CELLz を PLx の最後の CELL に設定する。CELLz とは、読出範囲の最後を規定する CELL である。

引数指定があれば、CELly を引数指定された CELL に設定し、CELLz を同じ引数指定された CELL に設定する。CELly、CELLz を、引数指定された CELL に設定しておくのは、引数で CELL が指定されている場合は、その CELL さえ読み出せばよいからである。

ステップ S 1 ~ ステップ S 19 は、CELly を構成する ACCESS UNIT の読み出しと、復号手順とを示す。この手順は、CELly の In 点ビデオフレームを含む ACCESS UNITv を TMAP から特定し(ステップ S 1)、ステップ S 2 ~ ステップ S 5 の判定を経てから ACCESS UNITv の読み出しを BD-ROM ドライブ 1 に命じて(ステップ S 6)、ステップ S 7 ~ ステップ S 8 の判定を経てから、ACCESS UNITv に含まれるビデオフレームのデコードを行うようビデオデコーダ 4 に指示して(ステップ S 9)、ACCESS UNITv を次の ACCESS UNIT に設定する(ステップ S 10)というものである。

ステップ S 2 は、ACCESS UNITv が属するセグメントは、未読出であるか否かを判定するものである。ACCESS UNITv が属するセグ

メントにおいて、ACCESS UNIT<sub>v</sub>が初めて読み出される ACCESS UNIT  
であるなら(ステップ S 2 で Yes)、ACCESS UNIT<sub>v</sub> が属するセグメ  
ントをセグメント i に設定し(ステップ S 1 1)、TMAP を参照して、  
セグメント i に先行するインターリーブユニット i を読み出して  
5 から(ステップ S 1 2)、ACCESS UNIT<sub>v</sub> の読み出しを BD ドライブ  
1 に命じる(ステップ S 6)。これにより、ACCESS UNIT<sub>v</sub> は、その  
先行するインターリーブユニットとともに BD から読み出される  
ことになる。

ここで ACCESS UNIT を含むセグメントはファイル「XXX.M2TS」  
10 に格納されており、これに先行するライブラリはファイル  
「ILUsss.CLASS」に格納されている。セグメント及びライブラリ  
はそれぞれ別々の 2 つのファイルに格納されているので、再生制  
御エンジン 1 2 は、ファイル「ILUsss.CLASS」からの読み出しと  
「XXX.M2TS」からの読み出しとを連続して行う必要がある。

15 ファイル「ILUsss.CLASS」からの読み出しは、ファイル  
「ILUsss.CLASS」をオープンし、オープンされたファイル  
「ILUsss.CLASS」の先頭位置を指定するようファイルポインタを  
設定してから、以降、ファイル「ILUsss.CLASS」の全体長を設定  
した READ コマンドを BD ドライブ 1 に発行することでなされる。  
20 このコマンド発行により、ファイル「ILUsss.CLASS」は再生装置  
200 のメモリ 1 4 に読み出されることになる。

ファイル「XXX.M2TS」からの読み出しは、ファイル「XXX.M2TS」  
をオープンし、オープンされたファイル「XXX.M2TS」のうち、  
ACCESS UNIT<sub>v</sub> の先頭位置を指定するようファイルポインタを設  
25 定してから、以降、ACCESS UNIT<sub>v</sub> の全体長を設定した READ コマ  
ンドを BD ドライブ 1 に発行することでなされる。このコマンド  
発行により、ACCESS UNIT<sub>v</sub> は再生装置 200 のトラックバッファ  
2 に読み出されることになる。

ファイル「ILUsss.CLASS」を BD ドライブ 1 からの読み出した  
30 段階で、ファイルポインタはファイル「ILUsss.CLASS」(インタ

ーリーブユニット)の終了位置を指している。ACCESS UNIT<sub>v</sub>を含むセグメントはインターリーブユニットの直後に位置しているので、光ピックアップの位置を僅かにずらすだけでACCESS UNIT<sub>v</sub>の先頭位置を指定するようファイルポインタを設定することができる。

ファイル「ILUsss.CLASS」からの読み出しと「XXX.M2TS」からの読み出しとにより、光ピックアップはシーク動作を行うことなく、セグメントとこれに先行するインターリーブユニットとをまとめて読み出すことができる。

ステップS3は、メモリにあるどれかのインターリーブユニットの生存区間In点に現在の再生時点が到達したかのどうかの判定であり、もしどれかのインターリーブユニットの生存区間IN点に到達していれば(ステップS3でYes)、そのインターリーブユニット内のライブラリを利用可能にして(ステップS13)、ステップS6に移行する。

インターリーブユニット内のライブラリが利用可能とされる前は、アプリケーションプログラム(Javaモードの動的シナリオのことである)がメモリ14からJava仮想マシン内のワーク領域へのライブラリロードを命じたとしても、このロードを行わず例外終了する。これは、ライブラリがメモリ14に読み出されていない場合と同じである。インターリーブユニット内のライブラリが利用可能になれば、アプリからのロード命令に応じてJava仮想マシン内のワーク領域にライブラリを転送する。かかる転送にて、ライブラリはJava仮想マシンにて実行されることになる。

ステップS4は、ユーザ操作があったかどうかの判定であり、もしユーザ操作があれば、それがSTOP操作であるかを判定し(ステップS15)、もしSTOP操作であれば(ステップS15でYes)、本フローチャートの処理を終える。それ以外なら、ユーザ操作に対応する処理を行い、ステップS2～ステップS10からなるループ処理に戻る。



ステップ S 5 は、現在の再生時点がメモリにあるどれかのインターリーブユニットの生存区間 Out 点に到達したかの判定であり、もし到達していれば、そのインターリーブユニットをメモリから削除して(ステップ S 1 4)ステップ S 6 に移行する。

- 5     ステップ S 7 は、ACCESS UNIT<sub>v</sub> が In 点ビデオフレームを含んでいるかどうかの判定である。もし IN 点ビデオフレームを含んでいる場合は(ステップ S 7 で Yes)、In 点のビデオフレームから ACCESS UNIT<sub>v</sub> の最後のビデオフレームまでのデコードを行うようビデオデコーダ 4 に指示してから(ステップ S 1 6)、ステップ
- 10    S 9 に移行する。

- ステップ S 8 は、Out 点 <sub>v</sub> が CELL<sub>y</sub> の Out 点ビデオフレームを含んでいるかどうかの判定である。もし Out 点ビデオフレームを含んでいる場合は(ステップ S 8 で Yes)、ACCESS UNIT<sub>v</sub> の先頭ビデオフレームから Out 点ビデオフレームまでのデコードを行うよう
- 15    ビデオデコーダ 4 に指示してから(ステップ S 1 7)、ステップ S 1 8 の判定を行う。ステップ S 1 8 は、本フローチャートの終了判定であり、CELL<sub>y</sub> が CELL<sub>z</sub> になったかを判定している。もし
- ステップ S 1 8 が Yes なら本フローチャートを終了し、そうでないなら、CELL<sub>y</sub> を次の CELL に設定して(ステップ S 1 9)、ステップ
- 20    S 1 に戻る。以降、ステップ S 1 8 が Yes と判定されるまで、ステップ S 1 ~ ステップ S 1 9 の処理は繰り返される。以上が再生制御エンジン 1 2 の処理手順である。

- 図 2 7 は、図 2 5 に示した構成要素のうち、ライブラリに係るものを抜き出して描いた図である。本図では、「ILUsss.CLASS」としてメモリ 1 4 に読み出されたライブラリ又はプログラムが、アプリケーションによる利用又は呼出に至るまでの経路を模式的に示している。ライブラリ及びプログラムは、インターリーブユニットの形態でセグメントの前に記録されている。そのためセグメントと共に矢印 gy1 に示すようにメモリ 1 4 に読み出される。
- 25    矢印 gy2 は、ライブラリがアプリケーションにより利用されるに
- 30

至るまでの経路であり、メモリ 14 から Java モジュール 17 内のワーク領域にロードされる。このロードによりアプリケーションは、矢印 gy3 に示すようにライブラリ内の関数をコールすることができる。

5 矢印 (gy4) は、「ILUsss.CLASS」内のプログラムが呼び出されるまでの経路であり、メモリ 14 から Java モジュール 17 内のワーク領域にロードされる。このロードにより、「ILUsss.CLASS」内のプログラムは”アプリケーション”と対等な地位になり、Java モジュール 17 により実行されることになる。

10 Java 仮想マシンのワーク領域にロードされたライブラリ又はアプリケーションが、イメージデコーダ 9 に対する制御 (gy5) を行うものなら、イメージデコーダ 9 により伸長処理が行われ、イメージプレーン 8 に伸長後のイメージが格納される。一方ビデオデコーダ 4 にはトラックバッファ 2、デマルチプレクサ 3 を介して ACCESS UNIT が読み出されているので、ピクチャプレーン 5 に  
15 は非圧縮型のピクチャがえられることになる。イメージデコーダ 9 が得たイメージと、ビデオデコーダ 4 が得たピクチャとを加算器 10 が合成することにより、図 21 のような合成画像が得られる。

20 以上のように本実施形態によれば、セグメント再生と同期実行すべきライブラリはインターリーブユニットという形でセグメントの前に配置されており、セグメントの再生時にあたっては光ピックアップによる読み取り位置をセグメントの先頭位置からずらせば、セグメントを、同期実行すべきライブラリと共に読み  
25 出すことができる。同期処理に必要なライブラリは、セグメントの読み出し時にセグメントと併せて BD-ROM から読み出せば足り、セグメントとの同期処理が済めば、メモリから削除すればよいので、デジタルストリーム再生の全区間を通じてライブラリを常駐しておく必要はない。デジタルストリームとの同期処理を行うべき  
30 ライブラリが複数あっても、個々のプログラムの実行が終わる

度に、ライブラリをメモリから回収するという原則でリソース管理を行えば、セグメント再生との同期時に、プログラムを格納し得るだけのメモリを再生装置に実装しておけばよい。必要最低限のメモリ規模で、デジタルストリーム再生と、様々なライブラリとの同期処理を実現することができる。

## (第2実施形態)

第1実施形態は、インターリーブ記録の客体はライブラリプログラムであったが、第2実施形態のインターリーブ記録の客体は、表示用データである。表示用データには、イメージ系データ、文書系データ、描画系データがある。イメージ系データには、JPEG, GIF, PNG, MNG データ等があり、イメージデコーダ9がダイレクトに伸長処理を行うことにより、表示可能となるデータをいう。文書系データには、HTML/SMIL, XML 文書, テキスト文書があり、BROWSER モジュール18により解釈されて、イメージデコーダ9が解釈結果に基づく表示制御を受けることにより表示可能となるデータをいう。

描画系データとは、いわゆる draw 系エディタを対象としたデータであり、立体形状や図形を、座標、ベクトル、曲率等で表現しているデータである。Java モジュール17により解釈され、イメージデコーダ9が解釈結果に基づく制御を受けることにより表示可能となるデータをいう。

インターリーブユニットたる表示用データは、セグメント再生と同期して表示される。ここでの同期表示は、セグメントの再生中に JPEG, GIF, PNG, MNG データを表示して合成したり、HTML/SMIL 文書, XML 文書, テキスト文書を別ウィンドウに表示させることを含む。HTML/SMIL 文書, XML 文書, テキスト文書は、ブラウザモードのアプリケーションプログラムにより利用され、JPEG, GIF, PNG, MNG データは Java モードのアプリケーションプログラムにより利用される。Java モードのアプリケーションプログラムは、AV ストリームの再生中において特定のシーンが出現した

場合、キャッシュメモリから Java 仮想マシン内ワーク領域へのロードを命じてくるだろうから、同期表示すべき期間より前に、表示用データをキャッシュメモリに読み出しておく必要がある。同様にブラウザモードのアプリケーションプログラムも、AV ストリームの再生中において特定のシーンが出現した場合、アプリケーションプログラムの利用を求めてくるだろうから、同期表示すべき期間より前に、表示用データをキャッシュメモリに読み出しておく必要がある。そこで第 2 実施形態では、表示用データを同期させたい期間から生存区間を求めて、生存区間の IN 点にあたる位置に、表示用データたるインターリーブユニットをインターリーブ記録する。こうすることでインターリーブユニットは、同期表示される期間の直前に再生装置のキャッシュメモリに読み出されることになる。

図 28 は、Java 言語が対象とするレイアモデルにおいて、かかるデータが何処に位置するかを示す図である。本図に示すようにセグメント再生と同期表示されるべきデータは、Native 描画系の上位層に配置される。

BD-ROM への表示用データの記録にあたって、表示用データはライブラリと同様 BD-ROM にインターリーブ記録される。

図 29 は、表示用データを格納したインターリーブユニットの構成を示す図である。本図におけるヘッダは、第 1 実施形態に示したロケータ、生存区間 IN 点情報、生存区間 OUT 点情報に加え、ファイルの「File Name」「Start\_address」言語属性(language\_type)」、AV ストリームの再生時間軸におけるファイルの「表示開始時刻情報」、「表示終了時刻情報」、画面における「表示座標情報」、表示エフェクトをかけるための「エフェクト情報」、「強制出力フラグ」をもっている。

「表示開始時刻情報」、「表示終了時刻情報」をヘッダにもたせているのは、インターリーブユニットに含まれる表示用データを、

AV ストリームを構成する動画データの再生と同期させて表示さ

せるには、表示用データの表示タイミングが必要となるからである。また生存区間とは別に「表示開始時刻情報」、「表示終了時刻情報」を定めているのは、メモリ上にインターリーブユニットを読み出しつつも、表示させたくないケースや、メモリ上にインターリーブユニットを配置しておいて表示用データの表示／非表示を繰り返すケースがあり、これらへの対処を念頭に置いているからである。

「言語属性(language\_type)」をヘッダにもたせているのは、表示用データが文章、会話に関する場合、どの言語で記述されているかを明示しておく必要があるためである。

「表示座標」をヘッダにもしているのは、動画データとの合成を想定した場合、背景たる動画データの絵柄によっては、表示用データが非常に見づらくなってしまう場合があり、見易い位置に表示用データを表示させるためである。

「エフェクト情報」をヘッダ内に設けておくのは、表示用データの表示開始時や表示終了時に、フェードインあるいはフェードアウトなどのエフェクトをかけたいとの要望に応えるためである。

「強制出力フラグ」をヘッダ内に設けておくのは、字幕を表示しないとユーザーが選択していた場合でも強制的に表示させたいという要望に応えるためである。本実施形態において表示用データは、図29に示したヘッダが付されてインターリーブユニットの形態で、セグメントの前に記録される。以上が本実施形態におけるインターリーブユニットの改良である。続いて本実施形態に係る再生装置の改良について説明する(イメージデコーダ9、スイッチャ15)。

スイッチャ15は、BDドライブ1の光ピックアップがインターリーブユニットが記録されているセクタ列から情報を読み取る度に、セクタからの読み取り情報をメモリ14に書き込む。読み取り情報のメモリ14への書き込みを繰り返せばインターリー

プユニットがメモリ 14 上に得られることになる。

5 イメージデコーダ 9 は、メモリ 14 に格納されたインターリーブユニットのヘッダを参照し、現在の再生時点が何れかのインターリーブユニット内の表示用データの表示開始時刻に到達すれば、インターリーブユニット内の表示用データを描画してイメージプレーン 8 に書き込む。一方、現在の再生時点が何れかのインターリーブユニット内の表示用データの表示終了時刻に到達すれば、イメージプレーン 8 に書き込まれた表示用データを消去する。

10 図 30 は、インターリーブユニットとしてメモリ 14 に読み出されたイメージ系データ、文書系データ、描画系データが表示に至るまでの経路を示す図である。矢印 my1 は、イメージ系データが表示に至るまでの経路を模式的に示す。イメージ系データは、インターリーブユニットの形態で BD-ROM からメモリ 14 に読み  
15 出され(gy1)、メモリ 14 からダイレクトにイメージデコーダ 9 に供給される(my1)。このイメージデコーダ 9 がデコード処理を行うことにより、表示に至る。

矢印 my2, my3 は、文書系データが表示に至るまでの経路であり、文書系データは、メモリ 14 から一旦 BROWSER モジュール 18 に  
20 読み出される(my2)。BROWSER モジュール 18 は、文書系データの解釈を行い、その解釈結果を出力し(my3)、解釈結果に基づきイメージデコーダ 9 に対する表示制御(ky1)を行うことにより、文書系データは表示に至る。

矢印 my4, my5 は、描画系データが表示に至るまでの経路である。  
25 描画系データは、メモリ 14 から一旦 Java モジュール 17 内のワーク領域にロードされる(my4)。Java モジュール 17 は、描画系データの解釈結果を出力し(my5)、解釈結果に基づきイメージデコーダ 9 に対する表示制御(ky2)を行うことにより、描画系データは表示に至る。

30 以上のように本実施形態によれば、表示用データをインターリ

ープ記録するので、セグメントの読み出しと共に表示用データを再生装置内のメモリに読み出すことができ、セグメント再生と、表示用データの表示との同期を容易に実現することができる。表示用データのための情報をインターリーブユニットのヘッダに  
5 設けておくことにより、インターリーブユニットの本体部にあたる表示用データの形式を変えることなく、AV ストリーム再生と同期して、各種データを表示させることができる。

### (第3実施形態)

第1実施形態では、プログラムの生存区間の開始点より前にインターリーブユニットを配置することでインターリーブユニットとプログラムとの同時読み込みを実現した。しかし第1実施形態では、生存区間の途中位置へと頭出しを行う場合でも、インターリーブユニット先頭から読み出しを行わねばならず、読み出しに長い時間がかかってしまう。BD-ROMにおいては、プレイリスト  
10 情報による再生経路の定義が可能であり、再生経路に従った再生にあたっては、AV ストリームを先頭から再生してゆくことは稀で、AV ストリームの途中から再生を開始してゆくことが頻繁に起こり得る。

AV ストリームやセグメントの途中からの頭出しが頻繁に発生するのなら、セグメントの前にインターリーブユニットを置くという配置では好適なインターリーブユニット読み出しを実行し得ない。そこで本実施形態は、インターリーブユニットに冗長性をもたせることを提案する。どのような冗長性かというのと、あるセグメントに先行するインターリーブユニットと同じもの(複製物)  
20 物)を、組み込んでおくというものである。複製物を組み込む箇所は、インターリーブユニットに含まれるライブラリの生存区間の途中にあたる位置である。図31は、セグメント内部のどこに、インターリーブユニット複製物を組み込むかを示す図である。本図において AV ストリームは、ライブラリ #x の生存区間の途中にあたる位置でセグメント i と、セグメント i+1 とに分割されてい  
30

る。そしてセグメント  $i$  とセグメント  $i+1$  との間にインターリーブユニットが配置されている。セグメント  $i$  とセグメント  $i+1$  との間のインターリーブユニットは、ライブラリ # $x$  を含んだものであり、セグメント直前に位置するインターリーブユニットの複製物である。つまりセグメントの前にあるインターリーブユニットの複製物が、ライブラリ # $x$  の生存区間の途中にあたる位置に存在しているのである。

図 3 2 は、セグメント  $i$  の内部に対して頭出しを行う場合、プログラムがどのように読み取られるかを示す図である。図中の (1) は、セグメント  $i$  内部に対する頭出し位置を示す。

インターリーブユニットはセグメント  $i$  の前に存在するので、本来ならセグメント  $i$  の前 ((2) にあたる位置) から光ピックアップによる読み出しを行わねばならない。しかし本実施形態では、プログラムの生存区間の途中にあたる位置に、インターリーブユニット複製物が存在するから、図 3 2 における頭出し位置 (1) から (3) に示すように読み出しを行えば、頭出し位置 (1) からインターリーブユニット複製物を読み出すことができる。ライブラリ # $x$  の生存区間の途中に、そのライブラリを含むインターリーブユニット # $x$  の複製物が記録されるので、生存区間途中に位置する ACCESS UNIT から頭出しをする場合、間近のインターリーブユニット複製物から、読み取りを行うことで AV ストリーム読み出しを開始することができる。セグメントに先行するインターリーブユニットまで光ピックアップを移動させなくてもよいので、デジタルストリームの頭出し位置からの読み出しやライブラリを用いた処理が容易になる。

続いて、図 2 2 ～図 2 4 に示したような 4 つのライブラリを、1 つの AV ストリームに組み込む場合の改良について説明する。図 3 3 は、図 2 2 に示した 4 つのライブラリ #1, #2, #3, #4 と、セグメントとの関係を示す図である。セグメント 1/5 ～セグメント 5/5 のうち、セグメント 2/5 ～セグメント 5/5 の境界は、図 2 2 (a)



に示したライブラリ#1,#2,#3,#4の生存区間のIn点と一致している。しかしセグメント1/5と2/5との境界は、ライブラリ#1の生存区間の途中にある。セグメント2/5の前にインターリーブユニット#1の複製物が配置され、以降セグメント3/5、セグメント4/5、セグメント5/5の前にインターリーブユニット#2、インターリーブユニット#3、インターリーブユニット#4が配置されている。セグメント2/5の前に、インターリーブユニット#1の複製物が配置されているので、セグメント2/5に含まれるACCESS UNITへの頭出しが生じた場合、セグメント2/5に先行するインターリーブユニット#1の複製物から光ピックアップによるスキャンを行えば、セグメント1/5に先行するインターリーブユニット#1まで光ピックアップを移動させなくても、インターリーブユニット#1をメモリに読み出すことができる。

以上のように本実施形態によれば、AVストリームを構成するACCESS UNITのうち、CELLのIN点等頭出しが多く発生しうる位置の後にインターリーブユニット複製物を配置しておけば、セグメント前に配置されているインターリーブユニットまでディスクシークを行わなくとも、アプリケーションプログラムによるライブラリ利用が可能となる。PL情報に基づく再生を行うため頭出しが多く発生したとしても、ディスクシークが最低限となり、スムーズな読み取りが可能となる。

#### (第4実施形態)

第1実施形態ではプログラム生存区間の開始点より前に、第3実施形態ではプログラム生存区間の途中にインターリーブユニット(若しくはその複製物)を配置していた。しかし何れの実施形態においても、デジタルストリームの逆方向再生が行われる場合、再生装置はインターリーブユニット読み出しを好適に行うことはできない。何故なら逆方向再生は、AVストリームの再生時間軸を未来から過去方向に遡ってゆくからである。逆方向再生をスムーズに行わせるため、本実施形態では、インターリーブユニット

に冗長性をもたせる。どのような冗長性かという、あるセグメントに先行するインターリーブユニットと同じものを、そのセグメントの後ろに組み込んでおくというものである。

図 3 4 (a) は、インターリーブユニット、セグメント、インターリーブユニット複製物の配置を示す図である。図 3 4 において  $i$  番目に再生されるセグメント  $i$  の前には、この再生時間内に生存区間をもつインターリーブユニット  $i$  が配置されている。セグメント  $i$  は、ライブラリ # $x$  の生存区間  $I_n$  点を含み、インターリーブユニット  $i$  は、ライブラリ # $x$  を含む。更にこのセグメント  $i$  の後には、セグメント  $i$  の前に配置されていたインターリーブユニットの複製物が配置されている。

図 3 4 (b) は、図 3 4 (a) に、 $i+1$  番目のセグメントを書き加えた図である。図 3 4 (a) 同様、セグメント  $i$  の後ろには、インターリーブユニット  $i$  の複製物が配置され、その後ろにはインターリーブユニット  $i+1$  が配置されている。かかるインターリーブユニット複製物を置くことの技術的意義は、逆方向再生を効率的に行うためである。

インターリーブユニット複製物がある場合と、無い場合とで逆方向再生がどのように行われるか対比して説明する。図 3 5 は、インターリーブユニット複製物がある場合に逆方向再生がどのように行われるかを示す図である。逆方向再生は、各セグメントに含まれる ACCESS UNIT を、後ろのものから順に読み出していかねばならない。図 3 5 の矢印 ①②③④ は、セグメントに ACCESS UNIT #8 が存在し、セグメントに ACCESS UNIT #4~#7 が存在する場合のディスクシークを示す。最初の ① は、セグメント  $i+1$  内の ACCESS UNIT #8 に属するピクチャデータを、先頭から読み出すというディスクシーク、

② は、インターリーブユニット  $i$  を読み出すというディスクシーク、③ は、ACCESS UNIT #7 を読み出すとのディスクシークである。図中のロングシーク 1 は、 $i+1$  番目のセグメント内の ACCESS

UNIT#8を読み出してから、 $i$ 番目のセグメントの直前にあるインターリーブユニット  $i$  をアクセスするまでの長いシーク動作を示す。ロングシーク 2 は、インターリーブユニット  $i$  を読み出してから、セグメント内の ACCESS UNIT#7 を読み取るまでの長いシーク動作を示す。

$i+1$  番目のセグメントを全て読み出してから、インターリーブユニット  $i$  までジャンプし(ロングシーク 1)、インターリーブユニット  $i$  から  $i$  番目のセグメントの最後の ACCESS UNIT をジャンプせねばならないので(ロングシーク 2)、シーク動作に無駄が多い。

図 3 6 は、インターリーブユニット複製物が配置された BD-ROM におけるシーク動作を示す図である。本図と図 3 5 との違いは、 $i+1$  番目のセグメントの前に、インターリーブユニット  $i+1$  が配置され、インターリーブユニット  $i+1$  の前にインターリーブユニット  $i$  の複製物が配置されている点である。シーク①は、 $i$  番目のインターリーブユニットの複製物と、 $i+1$  番目のインターリーブユニットと、セグメント  $i+1$  内の ACCESS UNIT とを読み出すディスクシークであり、シーク②は、 $i$  番目のセグメント内の ACCESS UNIT#7 を読み出すディスクシーク、シーク③は、 $i$  番目のセグメント内の ACCESS UNIT#6 を読み出すディスクシークである。

インターリーブユニット  $i$  の複製物が、インターリーブユニット  $i+1$  の直前に配置されているため、逆方向再生ではシーク①にて、 $i+1$  番目のセグメント内の先頭 ACCESS UNIT と共に、インターリーブユニット  $i$  が読み出される。セグメントの読み取りと共に、インターリーブユニットが読み取られるので、図 3 6 ではロングシークが不要となり、即座にセグメント  $i$  についてのインターリーブユニット  $i$  が読み取られることになる。以上のようにインターリーブユニットの複製物を、各セグメントの後ろ側に配置することで、逆方向再生時のロングシークの発生を回避することができる。

続いて、図 2 2 ～ 図 2 4 に示したような 4 つのライブラリを、  
1 つの AV ストリームに組み込む場合の改良について説明する。図  
3 7 は、図 2 2 に示した 4 つのライブラリ #1, #2, #3, #4 と、セグ  
メントとの関係を示す図である。つまり本実施形態では、セグ  
5 メント 1/4 の後ろ、インターリーブユニット #2 の前にインターリー  
ブユニット #1 の複製物が記録され、セグメント 2/4 の後ろ、イン  
ターリーブユニット #3 の前にインターリーブユニット #2 の複製  
物が、セグメント 3/4 の後ろ、インターリーブユニット #4 の前に  
10 インターリーブユニット #3 の複製物が記録されることになる。か  
かる複製物の配置により、逆方向再生におけるロングシークの発  
生を回避することができる。

以上が本実施形態に係る光ディスク側の改良である。続いて本  
実施形態に係る再生装置側の改良について説明する。図 3 6 に示  
したディスクアクセスは、逆方向再生時に再生制御エンジン 1 2  
15 が図 3 8 のフローチャートに基づく処理を行うことにより実現  
される。図 3 8 のフローチャートは、図 2 6 のフローチャートに  
おいてユーザにより逆方向再生操作が命じられた際、実行される  
ものである。

ステップ S 3 2 ～ ステップ S 4 0 は、CELLy を構成する ACCESS  
20 UNIT の読み出しと、復号手順とを示す。この手順は、ステップ S  
3 2 ～ ステップ S 3 5 の判定を経てから ACCESS UNITv の読み出  
しを BD-ROM ドライブ 1 に命じて (ステップ S 3 6)、ステップ S  
3 7 ～ ステップ S 3 8 の判定を経てから、ACCESS UNITv に含まれ  
るビデオフレームのデコードを行うようビデオデコーダ 4 に指  
25 示して (ステップ S 3 9)、ACCESS UNITv を次の ACCESS UNIT に設  
定する (ステップ S 4 0) というものである。

ステップ S 3 2 は、ACCESS UNITv が属するセグメントは、未読  
出であるか否かを判定するものである。もし未読出であるなら、  
ACCESS UNITv が属するセグメントは 2 番目以降であるかを判定し  
30 (ステップ S 4 2)、2 番目以降であるなら (ステップ S 4 2 で Yes)、

ACCESS UNITが属するセグメントをセグメント  $i+1$  に設定して(ステップ S 4 3)、TMAP を参照してセグメント  $i+1$  に先行するインターリーブユニット  $i+1$  と、インターリーブユニット  $i$  とを読み出す(ステップ S 4 4)。1 番目なら(ステップ S 4 2 で No)、ACCESS  
5 UNITが属するセグメントをセグメント  $i$  に設定して(ステップ S 1 1)、TMAP を参照してセグメント  $i$  に先行するインターリーブユニット  $i$  を読み出す(ステップ S 1 2)。ここで、ACCESS UNIT $v$  と、これに先行する 2 つのインターリーブユニットとはそれぞれ別々の 3 つのファイルに格納されているので、Java モードのアプリケーションには、2 つのインターリーブユニットにあたる 2 つ  
10 のファイルを読み出すコマンド、XXX.M2TS を読み出す読出コマンドを連続発行する必要がある。インターリーブユニット複製物の先頭アドレスに光ピックアップを移動させさえすれば、光ピックアップの一回の読み取り操作で、セグメントとこれに先行する 2  
15 つのインターリーブユニットとをまとめて読み出すことができる。

ステップ S 3 3 は、図 2 6 のステップ S 3 と同じ処理であり、メモリにあるどれかのインターリーブユニットの生存区間  $In$  点に現在の再生時点が到達したかのどうかの判定であり、もしどれ  
20 かのインターリーブユニットの生存区間  $IN$  点に到達していれば、そのインターリーブユニット内のライブラリを利用可能にする。

ステップ S 3 5 は、図 2 6 のステップ S 5 と同じ処理であり、現在の再生時点がメモリにあるどれかのインターリーブユニットの生存区間  $Out$  点に到達したかを判定し、もし到達していれば、  
25 そのインターリーブユニットをメモリから削除する。

ステップ S 3 7 は、ACCESS UNIT $v$  が  $Out$  点ビデオフレームを含んでいるかどうかの判定である。もし  $Out$  点ビデオフレームを含んでいる場合は(ステップ S 3 7 で Yes)、ACCESS UNIT $v$  の先頭ビデオフレームから  $Out$  点ビデオフレームまでのデコードを行うよう  
30 うビデオデコーダ 4 に指示してから(ステップ S 4 5)、ステップ

S 3 9 に移行する。

ステップ S 3 8 は、ACCESS UNIT<sub>v</sub> が CELL<sub>y</sub> の In 点ビデオフレームを含んでいるかどうかの判定である。もし In 点ビデオフレームを含んでいる場合は(ステップ S 3 8 で Yes)、In 点のビデオ  
5 フレームから ACCESS UNIT<sub>v</sub> の最後のビデオフレームまでのデコードを行うようビデオデコーダ 4 に指示してから(ステップ S 4 6)、ステップ S 4 7 の判定を行う。ステップ S 4 7 は、本フローチャートの終了判定であり、CELL<sub>y</sub> が PL を構成する最初の CELL  
10 になったかを判定している。もしステップ S 4 7 が Yes なら本フローチャートを終了し、そうでないなら、PL を構成する 1 つ前の CELL に CELL<sub>y</sub> を設定し(ステップ S 4 8)、CELL の IN 点ビデオフレームを含む ACCESS UNIT を TMAP から特定してから(ステップ S 4 9)ステップ S 3 2 に戻る。以降、ステップ S 4 7 が Yes と判定されるまで、ステップ S 3 2 ～ステップ S 4 9 の処理は繰り返  
15 される。以上が本実施形態に係る再生制御エンジン 1 2 の処理手順である。

以上のように本実施形態によれば、デジタルストリームの再生時間軸を未来から過去に遡って行き、ライブラリの生存区間に到達した場合でも、不要なシークを行うことなく BD-ROM からメモリへのライブラリ読み出しを実現することができる。逆方向再生  
20 のような変則的な再生進行が行われる場合でも、メモリへのライブラリ読み出しを実現出来るので、AV 再生との同期を前提とした様々な処理を、逆方向再生時でも実現可能となる。

#### (第 5 実施形態)

25 第 5 実施形態は、PL の再生を行い、この再生に同期して様々な再生制御を行う Java モードのアプリケーションについて説明する。

図 3 9 は、本実施形態に係る BD-ROM のファイル構成を示す図である。本図において新規なのは、

30 YYY.Mark(PLMark), XXX.Mark(ClipMark)が追加されている点であ

る。

PLMark は、PL の再生中において、再生装置が拡張制御を行うべき区間を示す情報である。ファイル名「YYY.Mark」のファイルボディ YYY は、PLMark が対応している PL と同じ名称が使用される。つまり本図における PLMark のファイル名のファイルボディは YYY であるから、PL(YYY.PL)に対応していることを意味する。

ClipMark は、AV ストリームの再生中において、再生装置が拡張制御を行うべき区間を示す情報である。XXX.Mark のファイル名におけるファイルボディ「XXX」は、ClipMark が対応している AV ストリームと同じ名称が使用される。つまり本図における ClipMark のファイル名におけるファイルボディは「XXX」であるから、AV ストリーム(XXX.M2TS)に対応していることを意味する。

ClipMark と PLMark との違いは以下の通りである。ClipMark は、拡張制御を行うべき区間を AV ストリームに対して指定しているのに対し、PLMark は PL に対して指定している。

例えば、図 11 に示したように 1 つの AV ストリームに対し、2 つの PL 情報が定義されている場合に、ClipMark にて拡張制御を行うべき区間を指定すれば、この指定は、この AV ストリームを指定している 2 つの PL 情報の何れに対しても及ぶ。一方、PLMark にて拡張制御を行うべき区間を指定すれば、PLMark に対応する PL にのみ、拡張制御指定の効力は及ぶ。PLMark に対応する PL が PL#1 なら PL#1 のみに及び、PL#2 には及ばない。つまり、拡張制御を行うべき区間を ClipMark にて指定すれば、AV ストリームを含む PL の全てに対し ClipMark の効力が及び、拡張制御を行うべき区間を PLMark にて指定すれば、その PLMark に対応する PL に対してのみ ClipMark の効力が及ぶのである。これが ClipMark と PLMark との違いである。

本実施形態における拡張制御とは、再生装置にイベントを発生させるというものである。イベントを発生させるため、PLMark 及び ClipMark は、図 40 に示す共通のデータ構造をもつ。図 4

0 は、PLMark 及び ClipMark の共通構成を示す図である。本図に示すように PLMark は、イベント数 (Number)、個々のイベント (Event#1~Event#m) とから構成され、再生中に発生すべきイベントを定義する。個々のイベント (Event#) は、イベントの種類 (Type) と、イベントの ID(ID) と、イベントが発生する時刻 (Time) と、このイベントが有効となる時間帯の時間長 (Duration) を示す。

かかるデータ構造にて定義されるイベントには、TimeEvent、UserEvent というものがある。TimeEvent とは、プレイリストの再生時間軸において、現在の再生位置が予め定められた時点 T に到達した場合に発生するイベントである。UserEvent とは、現在の再生位置が予め定められた時間帯に到達し、この時間帯でユーザにより操作がなされた場合に発生するイベントである。

プレイリスト #1 の再生中に出現する TimeEvent を定義する場合、PLMark をどのように記述するかについて図 4 1 (a) を参照しながら説明する。図中の下段の時間軸は、PL#1 が再生される時間軸を示す。この時間軸において時刻 t1 から発生する TimeEventEx1 を定義するケースを想定する。この TimeEvent を定義する場合の PLMark の記述例が、図中の PLMarkmt1 である。本 PLMark は、Type の項目が "TimeEvent" と記述され、ID の項目に "Ex1"、Time の項目が "t1"、Duration の項目が "0" と記述されている。図中の矢印は、時刻 t1 の到来で生じる TimeEvent を示す。この TimeEvent の発生により、図中のイベントハンドラが駆動されることになる。

一方、再生時間軸上の時刻 t1 から、間隔 T で TimeEvent を発生させる場合の PLMark の記述例を図 4 1 (b) に示す。図 4 1 (b) が (a) と違うのは、PLMark の Duration 項目が "T" と記述されている点である。項目 Duration の記述により、時点 t1 の経過後、間隔 T 置きに TimeEvent を発生させることができる。

プレイリスト #1 の再生中において UserEvent を定義する場合、



PLMark をどのように記述するかについて図 4 2 を参照しながら説明する。図 4 2 の中段の時間軸は、PL#1 が再生される再生時間軸を示す。この時間軸上において現在の再生時点がどこであるかは、SPRM(10)に示されている。この時間軸において時刻 t1 から期間 T1 の間、有効となる UserEventEv1 を定義するケースを想定する。この UserEvent を定義した PLMark が、図中の PLMarkmt2 である。本 PLMark は、Type の項目が" UserEvent" と記述され、ID の項目に" Ev1"、Time の項目が" t1"、Duration の項目が" T1" と記述されている。本図におけるリモコン rml は、ユーザ操作を受け付ける機器であり、図中の矢印 uv1 は、リモコンの ENTER キーの押下で生じる U0 を示す。U0 が、時刻 t1 から期間 T1 の時間帯に発生すれば、U0 に基づき UserEventEv1 が発生する。この UserEvent により、図中のイベントハンドラが駆動される。以上が PLMark により定義されるイベントである。尚 ClipMark により定義されるイベントについての説明は省略する。何故なら、PLMark が PL 再生中に発生すべきイベントを定義しているのに対し、ClipMark は AV ストリーム再生中に発生すべきイベントを定義しているにすぎず、PLMark におけるイベント定義とあまり変わらないからである。

図 4 3 は、ClipMark、PL にてかかる TimeEvent、UserEvent が定義されている場合のインターリーブユニットの配置例を示す図である。本図において AV ストリームは、TimeEvent を発生すべき時点 t1 の直前で分割され、この位置にイベントハンドラ Ex1 を格納したインターリーブユニットが記録されている。

このように記録しておくことで、時点 t1 を含むセグメントと共に、イベントハンドラを含むインターリーブユニットがメモリに読み出されることになる。イベントハンドラがメモリに読み出されれば、アプリケーションプログラムからのロード命令に応じて Java 仮想マシン内のワーク領域に、イベントハンドラをロードすることができるので、再生時点が時点 t1 に達すれば、この

イベントハンドラを即座に駆動することができる。

以上が本実施形態に係る BD-ROM の改良である。続いて、本実施形態における再生装置の改良点について説明する。上述したような TimeEvent、UserEvent を発生させるには、再生制御エンジン 12 に図 44 に示すような処理手順を行わせればよい。

図 44 は、本実施形態に係る再生制御エンジン 12 の処理手順を示す図である。本フローチャートは、図 22 のフローチャートの改良であり、同じ処理手順のものは同一の参照符号を付すか或は、図示を省略している。本フローチャートにおいて新規なのは、CELL を構成する ACCESS UNIT を読み出し、復号するという一連の処理(図 26 のステップ S2 ～ステップ S10 のうち、ステップ S5 と S6 との間)に、ステップ S20、ステップ S22 という 2 つの判定ステップが挿入されている点である。ステップ S20 は、PL 再生において、現在の再生時点を発生時刻とした TimeEvent が、PLMark、ClipMark に定義されているかどうかを判定するステップである。ステップ S20 が Yes なら、TimeEvent を発生して(ステップ S21)、ステップ S6 に移行する。TimeEvent を駆動要件とするイベントハンドラは、インターリーブユニットの形態で BD-ROM に記録されており、セグメントと共に再生装置のメモリに読み出されるので、TimeEvent を駆動要件とするイベントハンドラが即座に駆動される。

ステップ S22 は、U0 が発生したかどうかを判定しているステップである。ステップ S22 が Yes なら、PLMark、ClipMark を参照し、その U0 を有効としている時間帯であるかを判定する(ステップ S23)。もしその時間帯であれば、UserEvent の発生して(ステップ S24)、ステップ S6 に戻る。UserEvent を駆動要件とするイベントハンドラは、インターリーブユニットの形態で BD-ROM に記録されており、セグメントと共に再生装置のメモリに読み出されるので、UserEvent を駆動要件とするイベントハンドラが即座に駆動される。

以上のように本実施形態によれば、AV ストリームの再生時間上において、イベントハンドラの駆動が予定されている位置に、イベントハンドラをインターリーブユニットの形態でセグメント間に組み込んでおくので、アプリケーションプログラムによりロードが命じられた場合、メモリから Java 仮想マシン内のワーク領域へのロードを好適に行うことができる。

尚、TimeEvent、ILVU の他、PL の再生に先立ち発生するイベント (Pre イベント)、PL の再生後発生するイベント (Poset イベント) を再生制御エンジン 12 に発生させ、これにより駆動されるイベントハンドラをインターリーブユニットに格納してもよい。この場合、プレイリスト情報における先頭の CELL 情報にて指定されている時点及びプレイリスト情報における最後の CELL 情報にて指定されている時点のインターリーブユニットに格納しておくことが望ましい。

本実施形態では、Mark 情報にて定義されるイベントについてのイベントハンドラを説明したが、インターリーブ記録されるイベントハンドラは、他のプログラムにより発生されるイベントにて駆動されてもよい。機器により発生されるイベントにて駆動されてもよい。機器により発生されるイベントとは、BD-ROM のオープン／クローズがなされたことを示すイベント、電源異常等、機器の状態を示すイベント等がある。

#### (第 6 実施形態)

第 1 実施形態、第 2 実施形態においてライブラリ及び表示用データの生存区間に関する情報は、インターリーブユニットのヘッダに直接記述されていた。これに対し第 6 実施形態ではライブラリ及び表示用データの生存区間に関する情報を、間接参照の形式で記述するものである。

図 45 は、インターリーブユニットに関する情報の間接参照がどのように行われるかを示す図である。

本図の第 1 段たる AV ストリームは、2 つのセグメントに分割さ

れて BD-ROM に記録されており、このセグメントの合間にインターリーブユニットが記録されている。このインターリーブユニットはライブラリ、表示用データを含んでいる。ここまでは第 1 実施形態と変わらない。しかしヘッダのロケータ、生存区間 In 点情報、生存区間 Out 点情報が ID に置き換えられている。一方、第 2 段目、第 3 段目における静的なシナリオには、インターリーブユニット統合情報がある。このインターリーブユニット統合情報には、AV ストリームに組み込まれた複数インターリーブユニットのそれぞれについて、生存区間の情報が記述される。このインターリーブユニット統合情報は、各インターリーブユニットの生存区間 In 点情報、生存区間 Out 点情報、インターリーブユニットのサイズを対応づけて示している。ヘッダに直接生存区間 In 点情報、生存区間 Out 点情報を記述せず、ヘッダに ID を記述しておき、この ID を介してインターリーブユニット統合情報から生存区間 In 点情報、生存区間 Out 点情報を取得できるようにしている。これが生存区間 In 点情報、生存区間 Out 点情報の間接参照である。

図 4 6 は、AV ストリームに組み込まれた 3 つのインターリーブユニットと、インターリーブユニット統合情報との対応を示す図である。AV ストリーム内に ID=1, 2, 3... をもったインターリーブユニットが組み込まれている場合、ID=1, 2, 3... がインターリーブユニット統合情報に記述され、これらの ID 毎に、インターリーブユニットの生存区間 In 点情報、生存区間 Out 点情報、サイズが記述される。各インターリーブユニットに付された ID は、インターリーブユニット自身がどの AV ストリームに属しているか (1)、その AV ストリームにおいて何個目のインターリーブユニットであるか (2) を一意に識別する。

かかるインターリーブユニット統合情報が静的なシナリオ内に存在することにより、再生制御エンジン 1 2 は AV ストリームに組み込まれた複数インターリーブユニットの集中管理を行う

ことができる。集中管理の具体的な内容としては、生存区間が重なり合う複数インターリーブユニットが存在する場合、メモリ 14 へのインターリーブユニット読み出しを制限するというインターリーブユニットの読み出し本数制限がある。

- 5      インターリーブユニット統合情報は、各インターリーブユニットの ID と対応づけて各インターリーブユニットのサイズを示しているの、各インターリーブユニットのサイズを積算してゆけば、生存区間が重なり合う複数インターリーブユニットを同時にメモリ 14 に格納しようとした場合、メモリ 14 上に存在するインターリーブユニットの総サイズがどれだけになるかを、計算しておくことができる。もし計算された総サイズが、メモリ 14 のメモリサイズより大きくなれば、生存区間が重なり合う複数インターリーブユニットの全てをメモリ 14 に読み出すのではなく、一部のインターリーブユニットのみをメモリ 14 に読み出す。かかる読み出し本数の制限により、インターリーブユニット同士の上書きが回避され、メモリ 14 上でのインターリーブユニット破壊を回避することができる。
- 10
- 15

- 20      以上のように本実施形態によれば、キャッシュメモリ上に読み出し可能なインターリーブユニット本数を、インターリーブユニットの読み出しに先立ち事前に見積もることができるので、インターリーブユニットの生存区間が重なっている場合のキャッシュメモリの容量不足を回避することができる。

- 25      自機のメモリ規模に応じてどれだけインターリーブユニットを読み出すことができるかを調整することができるので、BD-ROM プレーヤモデルの策定にあたっては、「どれだけの規模のキャッシュメモリを具備せねばならない」というメモリ規模の指針を数値化しなくてもよくなる。メモリ規模の数値化が不要になるため、再生装置のメモリ規模がどれだけにすればよいかという、メモリ規模設計の自由度が高まり、BD-ROM 再生装置の普及に貢献することができる。
- 30

尚インターリーブユニット統合情報は、静的なシナリオにおいて1つ設けておいたが、ストリーム管理情報の1つの情報要素として組み込んでおいてもよい。この場合、個々のインターリーブユニットがどのAVストリームに組み込まれているかという識別(1)が不要であり、個々のインターリーブユニットがAVストリームにおける何番目のインターリーブユニットであるかを識別できればよいので、IDのビット長を短くすることができる。またPL情報の1つの情報要素として組み込んでもよい。

(第7実施形態)

- 10 第1実施形態及び第2実施形態では、再生時間軸上の生存区間のIn点及びOut点を生存区間In点情報、生存区間Out点情報で記述するという表記により、生存区間を表現していた。これに対し第7実施形態では、間接的な表記を提案する。どのような表記かということ、CELL情報の番号、PL(プレイリスト)情報の番号、Mark
- 15 情報の番号、タイトルの番号、チャプターの番号、SkipPointの番号を用いて生存区間を表記するのである。図47は、第7実施形態に係るインターリーブユニットにて、生存区間がどのように表記されるかを示す図である。

- ここで『CELL情報』は、第1実施形態に示したようにAVストリーム上の1つの再生区間を指定するものである。インターリーブユニットのヘッダに、生存区間情報としてCELL情報の番号を記述しておくことにより”再生区間”をインターリーブユニットの”生存区間”として定義することができる。図中のインターリーブユニットヘッダにおけるCELL情報の番号が、Cell情報#1を示しているなら、図中のck1に示すようにCELL情報#1にて指定
- 20 される再生区間(図中のay1~ay2で示される区間)が生存区間になる。

- 『PL情報』は、第1実施形態に示したように1つ以上の再生区間から構成される再生経路を指定するものである。インターリーブユニットのヘッダに生存区間情報としてPL情報の番号を記述
- 30

すれば、PL 情報にて指定された再生経路を、インターリーブユニットの生存区間として定義することができる。

『Mark 情報 (ClipMark, PLMark)』とは、第 5 実施形態に示したように拡張制御を行うべき区間を指定するものである。インターリーブユニットのヘッダに、生存区間情報として Mark 情報の番号を記述しておくことで、その Mark 情報にて指定される区間をインターリーブユニットの生存区間として定義することができる。

『タイトル』とは、1 つ以上の再生経路と、動的シナリオとから構成される 1 つの映画作品のことである。このタイトルの番号を生存区間情報としてインターリーブユニットのヘッダに記述しておくことにより、映画作品といった広汎な範囲の生存区間を定義することができる。

『チャプター』とは、映画作品の「章」としてユーザに認識される単位であり、1 つ以上の CELL 情報にて規定される。このチャプターの番号を生存区間としてインターリーブユニットのヘッダに記述しておくことにより、映画作品の「章」という、わかり易い単位で生存区間を定義することができる。

『SkipPoint』とは、章のうち、ユーザがマーキングを行った位置をいう。かかる位置は、Mark 情報で定義される。この SkipPoint の番号を生存区間 In 点情報、生存区間 Out 点情報としてインターリーブユニットのヘッダに記述しておくことにより、SkipPoint というユーザにとってわかり易い単位で生存区間を定義することができる。

以上が本実施形態に係る生存区間の表記である。続いてかかる生存区間を有している場合の、ライブラリ及び表示用データのインターリーブ記録について説明する。CELL 情報、PL 情報、Mark 情報、タイトル、チャプター、SkipPoint の番号で生存区間が表現されているとはいえ、生存区間が AV ストリーム上に存在していることは、第 1 実施形態と同様である。そのためライブラリ及

び表示用データは、AV ストリームのうち、CELL 情報、PL 情報～SkipPoint により指定されている位置の直前に記録すれば良い。かかる位置の前後で AV ストリームを複数セグメントに分割しておき、かかるセグメント間にライブラリ及び表示用データを含む  
5 インターリーブユニットを記録しておけばよいのである。そうすれば CELL 情報、PL 情報～SkipPoint を用いた再生の開始にあたって、インターリーブユニットは、かかる AV ストリームを構成するセグメントと共にメモリ 14 に読み出されることになる。

ここで CELL 情報により指定される位置とは、CELL 情報の In  
10 点情報により指定される AV ストリーム上の位置であり、PL 情報により指定される位置とは、PL 情報における最初の CELL 情報の In 点情報により指定される AV ストリーム上の位置である。

タイトルにより指定される位置とは、そのタイトルを構成する  
15 1 つ以上の PL 情報のうち、最初の PL 情報の最初の CELL 情報の In 点情報により指定される AV ストリーム上の位置であり、Mark 情報により指定される位置とは、AV ストリーム上においてその Mark 情報の時刻情報で指定される位置、チャプター番号、SkipPoint 番号で指定される位置とは、AV ストリーム上において  
20 そのチャプター、SkipPoint により指定される AV ストリーム上の位置をいう。

以上が本実施形態に係る BD-ROM の改良である。生存区間が、PL 情報、CELL 情報、タイトルの番号で表現されているので、第 7 実施形態に係る再生装置の再生制御エンジン 12 は、以下のような処理を行う。

25 PL 情報、CELL 情報、Mark 情報、タイトル、SkipPoint の番号により再生が命じられた場合、本実施形態に係る再生制御エンジン 12 は、これらで指定されている位置にあたる ACCESS UNIT を特定し、TMAP におけるその ACCESS UNIT に対応する ACCESS UNIT エントリーを参照する。その ACCESS UNIT エントリーにインター  
30 リーブユニットのファイル名が記述されていれば、そのファイル



名の ACCESS UNIT と、ACCESS UNIT とを連続して BD-ROM から読み出す。

かかる連続読み出しは、以下の 6 つのタイミングにおいて、なされることになる。

- 5      1 つ目のタイミングとは、タイトルが一覧表示され、このメニューに対してユーザが選択操作を行ったタイミングである。タイトル選択がなされれば、再生制御エンジン 1 2 は、このタイトルにて指定される位置の ACCESS UNIT に、インターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを ACCESS UNIT と共に BD-ROM からメモリ 1 4 に読み出す。メモリ 1 4 におけるインターリーブユニットの格納は、そのタイトルの再生が終了するまで継続してなされる。タイトル再生が終了すれば、再生制御エンジン 1 2 はメモリ 1 4 からインターリーブユニットを削除する。
- 10      2 つ目のタイミングとは、ある PL 情報がカレント PL 情報として静的シナリオメモリ 1 1 に読み出されるタイミングである。再生制御エンジン 1 2 はそのカレント PL 情報にて指定される位置の ACCESS UNIT にインターリーブユニットが先行しているかのサーチを行い、もし先行していれば ACCESS UNIT と共に、そのインターリーブユニットを BD-ROM からメモリ 1 4 に読み出す。メモリ 1 4 におけるインターリーブユニットの格納は、カレント PL 情報による再生が終了するまで継続してなされる。再生制御エンジン 1 2 は現在の再生時点がカレント PL 情報を構成する最後の CELL 情報の Out 点に到達するのを待ち、到達すれば、インターリーブユニットをメモリ 1 4 から削除する。Java モードのアプリケーションプログラムは、上述した PL 情報の再生が行われている間、ロード命令を発行することにより、ライブラリをメモリ 1 4 から Java 仮想マシン内のワーク領域に供給することができる。
- 15      3 つ目のタイミングとは、カレント PL 情報内の CELL 情報が再生の対象(図 2 6 のフローチャートにおける CELLy)として指定さ
- 20
- 25
- 30

れるタイミングである。そのタイミングにおいて再生制御エンジン 1 2 は、その CELL にて指定される位置の ACCESS UNIT に、インターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを BD-ROM からメモリ 1 4 に読み出す。再生制御エンジン 1 2 は現在の再生時点が CELL 情報の Out 点に到達するのを待ち、到達すれば、インターリーブユニットをメモリ 1 4 から削除する。かかる処理により、Java モードのアプリケーションプログラムは、上述した CELL 情報の再生が行われている間、ロード命令を発行することにより、ライブラリをメモリ 1 4 から Java 仮想マシン内のワーク領域に供給することができる。

4 つ目のタイミングとは、現在の再生時点が Mark 情報に示される時点に達したタイミングである。再生制御エンジン 1 2 は、その Mark 情報により指定された位置にあたる ACCESS UNIT に、インターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを BD-ROM からメモリ 1 4 に読み出す。

5 つ目のタイミングとは、ユーザがチャプターメニューからチャプターを選ぶ操作を行ったタイミングである。チャプター選択操作を行えば、再生制御エンジン 1 2 はそのチャプターにより指定された位置にあたる ACCESS UNIT に、インターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを BD-ROM からメモリ 1 4 に読み出す。尚、現在の再生時点がチャプターの先頭に到達した場合もそのチャプターにあたる位置に、インターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを BD-ROM からメモリ 1 4 に読み出す。

6 つ目のタイミングとは、ユーザが SkipPoint を選ぶ操作を行ったタイミングである。SkipPoint を選ぶ操作を行えば、再生制御エンジン 1 2 はその SkipPoint により指定された位置にあた

る ACCESS UNIT にインターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを BD-ROM からメモリ 14 に読み出す。尚、現在の再生時点が SkipPoint に到達した場合もその SkipPoint にあたる ACCESS UNIT

5 にインターリーブユニットが先行しているかのサーチを行い、もし先行していればそのインターリーブユニットを BD-ROM からメモリ 14 に読み出す。

以上のように本実施形態によれば、CELL 情報、PL 情報の番号を用いて生存区間を定義しているので、インターリーブユニット

10 同士の生存区間の重なるの把握が容易となり、リソース管理を念頭においたインターリーブユニットの読み取り計画が立て易くなる。

尚、再生制御エンジン 12 によるサーチを容易とするため、各インターリーブユニットの生存区間情報は、第 1 実施形態におけるインターリーブユニット統合情報において統合管理しておく

15 ことが望ましい。こうすることにより、インターリーブユニット統合情報を参照するだけで全てのインターリーブユニットに対するサーチが可能になる。

また本実施形態では、PL 情報、CELL 情報、タイトル、チャプター、SkipPoint と、インターリーブユニットとを番号で対応づけたが、PL 情報、CELL 情報、タイトル、チャプター、SkipPoint を一意に識別できるものなら、どのような識別情報であってもよい。また第 6 実施形態に示したようなインターリーブユニットの識別情報を、PL 情報、CELL 情報に付与することで識別してもよい。

20

25

#### (第 8 実施形態)

第 1 実施形態、第 2 実施形態では、ロケータにファイルパスを記述することによりメモリ 14 における個々のライブラリ及びプログラムを識別していたが、第 8 実施形態はロケータにおける

30 ファイルパスの記述により、メモリ 14 上のライブラリ、表示用

データの配置を階層化しようという改良に関する。図 4 8 は、ロ  
ケータの記述によるプログラム又は表示用データの階層化配置  
を模式的に示す図である。本図の第 1 段目は、AV ストリームと、  
AV ストリームの合間にインターリーブ記録されたインターリー  
5 プユニットとを示す。第 2 段目は、各インターリーブユニットの  
内部構成と、ヘッダにおけるロケータの記述とを示す。第 3 段目  
は、メモリ 1 4 上のプログラム及び表示用データの配置を示す。  
第 2 段目において各インターリーブユニットのロケータは、  
「CM://Java/Image」,「CM://Java/Game」,「CM://Java/Character」  
10 と記述されているため、再生制御エンジン 1 2 は各インターリー  
プユニットのヘッダを参照した際、メモリ 1 4 上の Java ディレ  
クトリを生成し、このディレクトリに「Image」,「Game」,  
「Character」というファイルを生成してこれらファイルに、イン  
ターリーブユニット内のプログラム又は表示用データを配置し  
15 てゆく。図中の矢印 ty1,ty2,ty3 は、この配置を模式的に示す。  
ここで第 2 段目の生存区間情報を見ると、インターリーブユニッ  
トの生存区間は、インターリーブユニット 1 の生存区間は 1~6、  
インターリーブユニット 2 の生存区間は 3~5、インターリーブユ  
ニット 3 の生存区間は 0~10 なので、各インターリーブユニット  
20 の生存区間は互いに重複し合っている。互いに重なり合うライ  
ブラリ及び表示用データを、「Image」,「Game」,「Character」というフ  
ァイルに格納して 1 つの java ディレクトリの配下に配置するの  
で、Java ディレクトリをアクセスすることにより、Java モード  
のアプリケーションプログラムはインターリーブユニット内の  
25 ライブラリ及び表示用データを Java 仮想マシン内のワーク領域  
にロードすることができる。

#### (第 9 実施形態)

これまでの実施形態では、ライブラリや表示用データのインタ  
30 ーリーブ記録を行っていた。これに対し第 9 実施形態では、ライ

ブラリや表示用データを、データカルーセルと同様の形式で AV  
ストリーム内に組み込むという改良に関する。データカルーセル  
とは、対話的な放送の実現のために同一内容を繰り返すとい  
う放送方式である。BD-ROM は、放送されたデータを格納するもの  
5 ではないが、このデータカルーセルの格納形式に倣ってライブラ  
リ及び表示用データを格納するようにしている。

ここでデータカルーセルでの送信について説明する。データカ  
ルーセルの客体は複数に分割される。分割により得られた個々の  
分割部分は、DDB(Download Datablock)というブロックになり、  
10 DII(DownloadinfoIndicatin)と呼ばれる固有の制御情報と共に  
AV ストリーム内に組み込まれる。この DII は、DDB に先立つ位置  
に配置され、DDB のデータ長が記述される。

このデータカルーセル方式の客体が、ライブラリ及び表示用デ  
ータであると考えれば、ライブラリ及びデータはヘッダが付加さ  
15 れた上で、複数の DDB に分割される。そしてこれらの DDB は、AV  
ストリームのうち、生存区間 In 点にあたる位置の前に配置され  
る。こうすることで、生存区間 In 点にあたる ACCESS UNIT が再  
生装置により読み出されたタイミングで、ライブラリ及び表示用  
データにあたる DDB は、再生装置に読み出されるのである。

20 以上が、第 9 実施形態に係る AV ストリーム側の改良である。  
続いて本実施形態に係る再生装置側の改良について説明する。図  
4 9 は第 9 実施形態に係る再生装置の内部構成を示す図である。

AV ストリーム内にデータカルーセル形式のライブラリ及びデ  
ータが存在するため、インターリーブユニットと AV ストリーム  
25 とを分離するスイッチ 1 5 は再生装置から省かれている。メモ  
リ 1 4 には、デマルチプレクサ 3 からライブラリ及びデータが供  
給されるようになっている。本図の再生装置においてデマルチプ  
レクサ 3 は、AV ストリームに対して多重分離を行い、ビデオスト  
リーム、オーディオストリーム、副映像ストリーム、そしてデー  
30 タカルーセル形式のライブラリ及びデータを得る。このうちデー

タカーセル形式のライブラリ及びデータはメモリ 14 に格納される。データカーセル形式のライブラリ及びデータはメモリ 14 に格納されるので、Java モードアプリケーションからのロード命令に応じて、データカーセル形式のライブラリ及びデータはメモリ 14 から Java 仮想マシン内のワーク領域に読み出されることになる。

以上のように本実施形態によれば、Java モードのアプリケーションプログラムにて利用され得るライブラリ及び表示用データを、データカーセルと同様の形式で再生装置に供給するので、再生装置の内部構成をデジタル放送の受信装置と共通化することができる。これにより、デジタル放送の受信装置の機能を兼ねるような再生装置を構成することができる。

#### (第 10 実施形態)

第 10 実施形態は、ライブラリー表示用データをインターリーブ記録の形態で BD-ROM に記録しつつも、再生装置からスイッチャ 15 を省略する改良に関する。スイッチャ 15 がなくなったため第 10 実施形態に係るアプリケーションプログラムは、以下のようにインターリーブユニットの読み出しを行う。

インターリーブユニットを ACCESS UNIT と共に読み出す場合、アプリケーションプログラムは、ファイル「ILUsss.CLASS」からの読み出しと「XXX.M2TS」からの読み出しとを連続して行う。

ファイル「ILUsss.CLASS」からの読み出しは、ファイル「ILUsss.CLASS」をオープンし、オープンされたファイル「ILUsss.CLASS」の先頭位置を指定するようファイルポインタを設定する。ここまでは第 1 実施形態と同様である。この際アプリケーションプログラムは、ファイル「ILUsss.CLASS」の読出先となる領域を宣言しておく。そしてファイル「ILUsss.CLASS」の全体長を設定した READ コマンドを BD ドライブ 1 に発行する。ここでいう読出先領域とは、メモリ 14 のことであり、このコマンド

発行により、ファイル「ILUsss.CLASS」は再生装置 200 のメモリ 14 に読み出されることになる。

ファイル「XXX.M2TS」からの読み出しは、ファイル「XXX.M2TS」をオープンし、オープンされたファイル「XXX.M2TS」のうち、  
5 ACCESS UNIT<sub>v</sub> の先頭位置を指定するようファイルポインタを設定する。ここまでは第 1 実施形態と同様である。この際アプリケーションプログラムは、ファイル「XXX.M2TS」の読出先となる領域を宣言しておく。ここでの読出先領域とは、トラックバッファ 2 のことである。そして以降、ACCESS UNIT<sub>v</sub> の全体長を設定した  
10 READ コマンドを BD ドライブ 1 に発行する。このコマンド発行により、ACCESS UNIT<sub>v</sub> は再生装置 200 のトラックバッファ 2 に読み出されることになる。

以上のように本実施形態によれば、アプリケーションプログラムがファイルを読み出すにあたって、ファイル「ILUsss.CLASS」  
15 を読み出す場合の読出先と、ファイル「XXX.M2TS」を読み出す場合の読出先とを切り換えることにより、スイッチャ 15 を用いずにインターリーブユニット読み出しと、AV ストリーム読み出しとを実行することができる。

## 20 (第 11 実施形態)

本実施形態は、BD-ROM の製造工程に関する実施形態である。図 50 は、第 11 実施形態に係る BD-ROM の製造工程を示すフローチャートである。

BD-ROM の制作工程は、動画収録、音声収録等の素材作成を行う  
25 素材制作工程 S101、オーサリング装置を用いて、アプリケーションフォーマットを生成するオーサリング工程 S102、BD-ROM の原盤を作成し、プレス・貼り合わせを行って、BD-ROM を完成させるプレス工程 103 を含む。

これらの工程のうち、BD-ROM を対象としたオーサリング工程は、  
30 シナリオ編集工程 S201、素材エンコード工程 S202、多重

化工程 S 2 0 3、フォーマッティング工程 S 2 0 4 という工程とからなる。

シナリオ編集工程 S 2 0 1 とは、企画段階において作成された筋書きを再生装置が理解できる形式に変換する工程である。シナ  
5 リオ編集の結果は、BD-ROM 用の静的シナリオとして生成される。また、このシナリオ編集において、多重化を実現するため多重化パラメータの等も生成される。

素材エンコード工程 S 2 0 2 とは、ビデオ素材、オーディオ素材、副映像素材のそれぞれをエンコードして、ビデオストリーム、  
10 オーディオストリーム、副映像ストリームを得る作業である。

多重化工程 S 2 0 3 では、素材エンコードにより得られた、ビデオストリーム、オーディオストリーム、副映像ストリームをインターリーブ多重化して、これらを 1 本の AV ストリームに変換する。

15 一方、これらの工程と並行して、Java 言語によるプログラム(工程 S 2 0 4、S 2 0 5)が実行される。

Java アプリケーション制作工程 S 2 0 4 は、Java 言語を用いて Java アプリケーションを制作する工程である。

ライブラリ制作工程 S 2 0 5 は、ライブラリを制作する工程で  
20 ある。

フォーマッティング工程 S 2 0 6 は、工程 S 2 0 1 ~ S 2 0 5 にて得られた AV ストリーム、静的シナリオ、動的シナリオ、ライブラリを BD-ROM のフォーマットに適合させる。この適合にあたって、第 1 実施形態に示したように、AV ストリームを複数のセ  
25 グメントに分割し、各セグメントの合間にライブラリを記録するとの処理がなされる。

上述した工程のうち、工程 S 2 0 4 において、Java アプリケーションである動的シナリオは、Java 言語を用いた記述が可能であるから、通常のコンピュータ向けのソフトウェアを開発するのと同  
30 じ感覚で、開発することができる。よって本実施形態では、こ



のシナリオ制作の効率を高めることができるという効果がある。

(第 1 2 実施形態)

本実施形態では、BD における AV ストリームを、BD-R、BD-RE 等の記録型記録媒体にコピーする実施形態である。その際記録装置は、動的シナリオ、テーブルを WEB サイトからダウンロードする。このテーブルは、図 2 2 (a) に示したような AV ストリーム再生時間軸に対する、ライブラリ生存区間の割り付けを示す。テーブルには、生存区間割り付けが示されているので、第 1 実施形態に示したように生存区間 IN 点を基準にして AV ストリームを分割し、セグメントの合間にセグメント内に生存区間をもつライブラリを記録する。この記録にあたってライブラリはインターリーブユニットに変換しておく。

こうすることで、ライブラリが記録されていないバージョンの BD を、ライブラリや動的シナリオが記録されたバージョンにアップグレードすることができる。(備考)

上記実施形態に基づいて説明してきたが、現状において最善の効果が期待できるシステム例として提示したに過ぎない。本発明はその要旨を逸脱しない範囲で変更実施することができる。代表的な変更実施の形態として、以下 (A) (B) (C) ……のものがある。

(A) 第 2 実施形態では、1 つの表示用データのみインターリーブユニットに格納していたが、複数の表示用データを 1 つのインターリーブユニットに変換して BD-ROM に記録してもよい。図 5 1 は、複数の表示用データを格納したインターリーブユニットを示す図である。本図におけるイメージ系データ、文書系データ、描画系データは、それぞれ複数の表示用データであり、かかるインターリーブユニットを BD-ROM に記録しておくことにより、インターリーブユニットの一回の読み出しにて複数の表示用データをメモリに格納しておくことができる。尚、この場合、ヘッダには、インターリーブユニットに含まれるそれぞれのファイルについての情報を記述しておく必要がある。

(B)図 1 2 に示したレイアモデルにおいて、Java モードの上にブラウザモード及び MOVIE モードを配置してもよい。特に MOVIE モードでの動的シナリオの解釈や、動的シナリオに基づく制御手順の実行は、再生装置に対する負担が軽いので、MOVIE モードを  
5 Java モード上で実行させても何等問題は生じないからである。また再生装置や映画作品の開発にあたって、動作保証が 1 つのモードで済むからである。

更に 3 つのモードを設けず、Java モードだけで Java モードの処理を実行してもよい。Java モードでも PL の再生と同期した再生制御が可能になるから、強いて MOVIE モードを設けなくてもよいという理由による。更に動的シナリオにおける制御は、MOVIE  
10 モードだけでも、ブラウザモードだけでもよい。

(B)全ての実施形態では、本発明に係る光ディスクを BD-ROM として実施したが、本発明の光ディスクは、記録されるインターリーブユニットがあり、この特徴は、BD-ROM の物理的性質に依存するものではない。光ディスクなら、どのような光ディスクであつてもよい。例えば、

DVD-ROM, DVD-RAM, DVD-RW, DVD-R, DVD+RW, DVD+R, CD-R, CD-RW 等の光ディスク、PD, MO 等の光磁気ディスクであってもよい。

動的シナリオ、プレイリスト情報は、AV ストリーム及びストリーム管理情報と別々の記録媒体に記録されてもよい。そしてこれらを平行に読み出して、1 つの映画作品として再生させてもよい。

(C) 全ての実施形態における再生装置は、BD-ROM に記録された AV ストリームをデコードした上で TV に出力していたが、再生装置を BD-ROM ドライブのみとし、これ以外の構成要素を TV に具備させてもい、この場合、再生装置と、TV とを IEEE1394 で接続されたホームネットワークに組み入れることができる。また、実施形態における再生装置は、テレビと接続して利用されるタイプで  
25 あったが、ディスプレイと一体型となった再生装置であってもよ  
30

い。更に、各実施形態の再生装置において、処理の本質的部分を  
なす部分のみを、再生装置としてもよい。これらの再生装置は、  
何れも本願明細書に記載された発明であるから、これらの何れの  
態様であろうとも、各実施形態に示した再生装置の内部構成を元  
5 に、再生装置を製造する行為は、本願の明細書に記載された発明  
の実施行為になる。各実施形態に示した再生装置の有償・無償に  
よる譲渡（有償の場合は販売、無償の場合は贈与になる）、貸与、  
輸入する行為も、本発明の実施行為である。店頭展示、カタログ  
勧誘、パンフレット配布により、これらの譲渡や貸渡を、一般ユー  
10 ーザに申し出る行為も本再生装置の実施行為である。

(D)図26、図38、図44に示したプログラムによる情報処理は、ハードウェア資源を用いて具体的に実現されていることから、上記フローチャートに処理手順を示したプログラムは、単体で発明として成立する。全ての実施形態は、再生装置に組み込まれた態様で、本発明に係るプログラムの実施行為についての実施  
15 形態を示したが、再生装置から分離して、各実施形態に示したプログラム単体を実施してもよい。プログラム単体の実施行為には、これらのプログラムを生産する行為(1)や、有償・無償によりプログラムを譲渡する行為(2)、貸与する行為(3)、輸入する行為(4)、  
20 双方向の電子通信回線を介して公衆に提供する行為(5)、店頭展示、カタログ勧誘、パンフレット配布により、プログラムの譲渡や貸渡を、一般ユーザに申し出る行為(6)がある。

(E)図26、図38、図44のフローチャートにおいて時系列に実行される各ステップの「時」の要素を、発明を特定するための  
25 必須の事項と考える。そうすると、これらのフローチャートによる処理手順は、再生方法の使用形態を開示していることがわかる。各ステップの処理を、時系列に行うことで、本発明の本来の目的を達成し、作用及び効果を奏するよう、これらのフローチャートの処理を行うのであれば、本発明に係る記録方法の実施行為に該  
30 当することはいうまでもない。

(F)Java モードのアプリケーションは、コンピュータ・グラフィックスを描画するようなアプリケーションであったが、Java 言語で記述されるアプリケーションであればどのようなものであってもよい。例えば電子商取引(EC(Electronic Commerce))のクライアントアプリケーションであってもよい。映画作品の動画を交えながら商品案内を行うような Java モードのアプリケーションを実現することができるので、映画作品に関連するキャラクタービジネスを成功に導くことができる。また Java モードのアプリケーションは、ネット対戦型のオンラインゲームであってもよい。更に、Java モードのアプリケーションにてコンピュータ・グラフィックスにて表現されるキャラクターは、エージェントとしての処理を行うものであってもよい。エージェントとしてのキャラクターは、再生装置のヘルプ機能を実現するものであったり、ユーザに助言を行うものでもよい。

WebPage オブジェクトが WEB サイトから取得する情報は、WEB ページであってもよいし、画像データであってもよい。また、AV ストリーム、ストリーム管理情報、PL 情報であってもよい。また WebPage オブジェクトは、検索エンジンと連携して処理を行ってもよい。

更に、エンハンスドモードにおける記述言語は、C++や C#言語であってもよい。

(G)BD-ROM に記録するにあたって、AV ストリームを構成する各 TS パケットには、拡張ヘッダを付与しておくことが望ましい。拡張ヘッダは、TP\_extra\_header と呼ばれ、『Arrival\_Time\_Stamp』と、『copy\_permission\_indicator』とを含み 4 バイトのデータ長を有する。TP\_extra\_header 付き TS パケット(以下 EX 付き TS パケットと略す)は、32 個毎にグループ化されて、3 つのセクタに書き込まれる。32 個の EX 付き TS パケットからなるグループは、6144 バイト(=32×192)であり、これは 3 個のセクタサイズ 6144 バイト(=2048×3)と一致する。3 個のセクタに収められた 32 個の

EX 付き TS パケットを "Aligned Unit" という。

IEEE1394 を介して接続されたホームネットワークでの利用時において、再生装置 200 は、以下のような送信処理にて Aligned Unit の送信を行う。つまり送り手側の機器は、Aligned Unit に含まれる 32 個の EX 付き TS パケットのそれぞれから TP\_extra\_header を取り外し、TS パケット本体を DTCP 規格に基づき暗号化して出力する。TS パケットの出力にあたっては、TS パケット間の随所に、isochronous パケットを挿入する。この挿入箇所は、TP\_extra\_header の Arrival\_Time\_Stamp に示される時刻に基づいた位置である。TS パケットの出力に伴い、再生装置 200 は DTCP\_Descriptor を出力する。DTCP\_Descriptor は、TP\_extra\_header におけるコピー許否設定を示す。ここで「コピー禁止」を示すよう DTCP\_Descriptor を記述しておけば、IEEE1394 を介して接続されたホームネットワークでの利用時において TS パケットは、他の機器に記録されることはない。

(H) 第 12 実施形態では、AV ストリームにコピー世代情報 (CGI) を埋め込んで、AV ストリームの制限付きコピーを認めてもよい。コピー世代情報には、自由にコピー可能な旨を示す『copy free』と、1 世代の複製物の記録が許可されている旨を示す『one generation copy』と、複製物の更なる複製は禁止されている旨を示す『no more copy』と、一切のコピーを認めない旨を示す『never copy』とがある。つまり、コンテンツに埋め込まれたコピー世代情報が『copy free』や『one generation copy』を示している場合のみ、HD から DVD へのバックアップを行い、『no more copy』『never copy』を示している場合、バックアップは行わないのである。

(I) AV ストリームに権利管理情報は付加されている場合、第 12 実施形態では権利管理情報に定められた利用条件において、AV ストリームのコピーを行うべきである。

30 権利管理情報において回数指定で、コピーの許可条件が規定さ

れているなら、この条件内でコピーを行う。有効期間の指定(年月日の指定がでコピーの条件が規定されているのならこの条件内でコピーを行う。

(J)コピー処理に、移動やマイグレート、チェックアウトといったバリエーションが存在する場合、これらバリエーション毎に、  
5 定められた利用条件に基づきバックアップを行っても良い。移動とは、コピー元のコンテンツの削除が伴うコピー処理であり、複数記録媒体間でコンテンツを移動する場合に用いられる。

マイグレートとは、コピー先記録媒体において、利用条件情報  
10 を生成させることを前提にしたコピー処理である。

チェックアウトとは、回数制限されたコピーの一種であり、コピー回数を1回デクリメントした上でコンテンツのコピーが実行される。一般にいう回数制限付きのコピーと、チェックアウトとの違いは、コピー回数のインクリメントが可能である点である。  
15 コピー回数のインクリメントは、コピーにより記録媒体に記録されたコンテンツを再生不能にするという処理("チェックイン"と呼ばれる。)を行った後でなされる。

(K)各実施形態における AV ストリームは、DVD-Video 規格、DVD-Video Recording 規格の VOB(Video Object)であってもよい。  
20 VOB は、ビデオストリーム、オーディオストリームを多重化することにより得られた ISO/IEC13818-1 規格準拠のプログラムストリームである。また、各実施形態における AV ストリームを、"AVClip"と呼んでもよい。この場合、ストリーム管理情報は Clip 情報と呼ぶことになる。また AV ストリームにおけるビデオスト  
25 リームは、MPEG4 や WMV 方式であってもよい。更にオーディオストリームは、Linear-PCM 方式、Dolby-AC3 方式、MP3 方式、MPEG-AAC 方式であってもよい。

(L)各実施形態における Cell 情報は、時間情報により再生区間の開始点、終了点を指定していたが、BD-ROM における論理アドレスを用いて再生区間の開始点、終了点を指定してもよい。また、  
30

各実施形態における CELL を "PlayItem" と呼んでもよい。

(M) ストリーム管理情報における TMAP は、EP\_map と呼んでもよい。この場合、ACCESS UNIT の再生開始時刻を、ACCESS UNIT 先頭に位置するピクチャデータのタイムスタンプ (Presentation Time Stamp) で表現するのが望ましい。また ACCESS UNIT におけるアドレスは、PES パケットの連番 (SPN (Serial Packet Number)) で表現するのが望ましい。

(N) 再生装置の構成において、動的シナリオメモリ 21 には、カレントの動的シナリオのみを格納し、静的シナリオメモリ 11 には、カレントのストリーム管理情報、カレントの PL 情報のみを格納するとしたが、複数のシナリオ、ストリーム管理情報、PL 情報を予め格納しておいてもよい。こうすれば、これらのデータを BD-ROM から読み出すまでのタイムラグを短くすることができる。

(O) PL を構成する 2 以上の CELL を連続再生させるには、これらの CELL がシームレス接続されよう、加工を施しておくことが望ましい。

シームレス接続のための加工は、動画データにおいて先行する側の再生区間の終端部と、後続する側の再生区間の先端部とを複製することにより、予め複製部分を作成しておき、これらを再エンコードすることで、実現される。尚、シームレス接続のために作成された複製部分を、Bridge-Clip と呼んでもよい。

ここで終端部、先端部は、以下のように設定するのが望ましい。

つまり先行する VOB#x のうち先行再生区間の Out 点を含む ACCESS UNIT から、2 個先の ACCESS UNIT までを終端部とし、また後続する CELL 情報 #x+1 のうち後続再生区間の In 点を含む ACCESS UNIT を先端部とするのが望ましい。終端部及び先端部をこのように定める根拠は、同出願人の先行技術米国特許 USP, 6148,140 公報により記載されているので、詳細に関してはこの公報を参照されたい。

更に、シームレス接続のために作成された複製部分については、シームレス接続情報を設けておくことが望ましい。シームレス接続情報とは、最初のビデオフレームの再生開始時刻、最後のビデオフレームの再生終了時刻、オーディオギャップの開始時刻、オーディオギャップの時間長、オーディオギャップの位置情報を含む情報である。かかるシームレス接続情報が定義されていれば、最初のビデオフレームの再生開始時刻、最後のビデオフレームの再生終了時刻から、両区間のタイムスタンプの差(STC-Offset)を計算して、再生装置に設定することができる。また、これらオーディオギャップの情報を参照して、オーディオデコーダを制御すれば、1つの区間から別の区間への移行する際の音声の途切れを防止することができる。

(P)各実施形態における映画作品は、テレビ映画、ゲームソフト等、映像をもって表現されている全ての著作物を意味する。何故なら、各実施形態における映画作品は、ブラウン管や液晶での表示など、視覚的又は視聴覚的效果を生じさせる方法で表現され(i)、BD-ROMという有体物に、何らかの方法で結びつくことにより同一性を保ちながら存続しており(ii)、かつ再現が可能な状態におかれているので(iii)、映画の著作物の存在形式としての要件を満たすからである。

しかし本発明の本質的要素は、この映画作品であることに限定されるものではないので、本発明の動画データは、監視カメラの映像や家庭用ビデオカメラの映像であってもよい。

(Q)第2実施形態において描画系データは、コンピュータ・グラフィックスのデータであってもよい。かかるデータには、NURBS(Non Uniform Rational B-Spline)形式のデータ、ポリゴン形式がある。NURBSとは、ベジェ曲線の束であって(このようなベジェ曲線の束は B-Spline と呼ばれる。)、それぞれのベジェ曲線の曲率が不均一なものをいう。

ポリゴン形式とは、多面体近似により特定の立体形状を表現す



るよう規定されたデータ形式であり、アメリカ・オートキャド社が規定した Data eXchange Format(DXF) を初め、HRC, WAVEFRONT, IV, VRML 形式等のものが広く知られている。

また第2実施形態に示したイメージ系データを、テクスチャマッピングに利用してもよい。テクスチャマッピングとは、三次元形状の平面又は曲面に、静止画、ビットマップ等のテクスチャパターンを貼り付けて表示させる処理をいう。コンピュータ・グラフィックスの描画のため、OPEN-GL や Java3D 等のプログラムを、インターリブ記録しておいてもよい。

5 (R)各実施形態では、ユーザによる映画作品の選択操作を、リモコンから受け付けたが、再生装置のフロントパネルから受け付けてもよい。キーボード、タッチパネル、マウスやパッド、トラックボール等の入力機器にてユーザの指定を受け付けてもよい。この場合は、クリック操作、ドラッグ操作でこれらの操作を受け  
15 付けてもよい。

(S)各実施形態における映画作品は、アナログ放送で放送されたアナログ映像信号をエンコードすることにより得られたものでもよい。デジタル放送で放送されたトランスポートストリームから構成されるストリームデータであってもよい。

20 またビデオテープに記録されているアナログ／デジタルの映像信号をエンコードしてコンテンツを得ても良い。更にビデオカメラから直接取り込んだアナログ／デジタルの映像信号をエンコードしてコンテンツを得ても良い。他にも、配信サーバにより配信されるデジタル著作物でもよい。

25 (T)Java モジュール 17 は、衛星放送受信のために機器に組み込まれた Java プラットフォームであってもよい。Java モジュール 17 がかかる Java プラットフォームであれば、本発明に係る再生装置は、MHP 用 STB としての処理を兼用することになる。

更に携帯電話の処理制御のために機器に組み込まれた Java プラットフォームであってもよい。かかる Java モジュール 17 が  
30

かかる Java プラットフォームであれば、本発明に係る再生装置は、携帯電話としての処理を兼用することになる。

また BROWSER モジュール 18 は、MicroSoft 社の Internet Explore 等、パソコン組み込み型のブラウザソフトであってもよい。

#### 符号の説明

	1	BD ドライブ
	2	トラックバッファ
10	3	デマルチプレクサ
	4	ビデオデコーダ
	5	ピクチャプレーン
	6	オーディオデコーダ
	8	イメージプレーン
15	9	イメージデコーダ
	10	加算器
	11	静的シナリオメモリ
	12	再生制御エンジン
	13	プレーヤレジスタ
20	14	メモリ
	15	スイッチャ
	16	DVD ライクモジュール
	18	BROWSER モジュール
	17	Java モジュール
25	19	UO マネージャ
	21	動的シナリオメモリ
	200	再生装置
	300	テレビ
	400	リモコン

### 産業上の利用可能性

本発明に係る光ディスクは、ゲームソフトの動作環境を、映画作品を再生するための動作環境に近づけることができるので、より魅力ある映画作品を市場に供給することができ、映画市場や民生機器市場を活性化させることができる。故に本発明に係る光ディスク、再生装置は、映画産業や民生機器産業において高い利用可能性をもつ。

## 請 求 の 範 囲

1. デジタルストリームを記録した光ディスクであって、  
デジタルストリームは、 $n$  個のセグメントに分割されて光ディスクに記録されていて、

- 5  $n$  個のセグメントのうち、第  $i$  番目に再生されるものの前に、  
インターリーブユニットが記録されており、  
ここで  $i, n$  は、 $i < n$  を満たす整数であり、  
インターリーブユニットは、第  $i$  番目のセグメントの再生に同期するプログラム、又は、  
10 第  $i$  番目のセグメントの再生に同期して表示されるデータを含むことを特徴とする光ディスク。

2. インターリーブユニットは終了時点情報を含み、  
終了時点情報は、デジタルストリームの再生時間軸上における  
15 どの時点でメモリに読み出されたプログラム又はデータを削除するかを示す  
ことを特徴とする請求項 1 記載の光ディスク。

3. インターリーブユニットは開始時点情報を含み、  
20 開始時点情報は、インターリーブユニット内に含まれるプログラム又はデータの利用が、デジタルストリームの再生時間軸上におけるどの時点から、可能になるかを示す  
ことを特徴とする請求項 1 記載の光ディスク。

- 25 4. 第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの間には、前記インターリーブユニットの複製物が記録されている  
ことを特徴とする請求項 1 記載の光ディスク。

5. 第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの境界は、プログラム又はデータの生存区間の途中にあたる  
30

ことを特徴とする請求項 4 記載の光ディスク。

6. 第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの境界は、前記プログラムの生存区間の終了点より後にあたる

5 ことを特徴とする請求項 4 記載の光ディスク。

7.  $i$  番目のセグメントは、複数のアクセスユニットからなり、各アクセスユニットはイントラ符号化されたピクチャを有した動画データを含み、

10 光ディスクにはマップ情報が記録されており、

マップ情報は、 $i$  番目セグメントに先行するインターリーブユニットを、第  $i$  番目セグメントに属するアクセスユニットのアドレス又は再生時間と対応づけて示す

ことを特徴とする請求項 1 記載の光ディスク。

15

8. 前記  $i$  は 2 以上であり、

$i$  番目のセグメントと、 $i-1$  番目のセグメントとの境界は、デジタルストリームの再生時間軸において、プログラム又はデータの生存区間の開始点にあたる

20 の光ディスク。ことを特徴とする請求項 1 記載

9. 前記生存区間とは、デジタルストリームの再生時間軸上において、インターリーブユニット内のプログラム又はデータの利用が可能となる区間である、請求項 8 記載の光ディスク。

25

10. 前記プログラムは所定のイベントで駆動されるイベントハンドラであり、

イベントハンドラを含むインターリーブユニットは、デジタルストリームの再生時間上において、前記イベントが発生する時点

30 より前に記録されている、請求項 1 記載の光ディスク。

1 1. 所定のイベントとは、現在の再生時点が再生時間軸上の  
所定の時点に到達したことを示すイベント、

再生時間軸上の所定の時間帯にてユーザ操作がなされたこと  
5 を示すイベント、 再生経路による再生に先立ち発生するイベン  
ト、

再生経路による再生後に発生するイベント、

再生装置が発生するイベント、

他のプログラムにより発生されるイベントの何れかである、請  
10 求項 10 記載の光ディスク。

1 2. 光ディスクには、インターリーブユニット統合情報が記  
録されており、

インターリーブユニット統合情報は、光ディスクに記録された  
15 複数インターリーブユニットのそれぞれの識別情報を、プログラ  
ム又はデータのサイズ及び生存区間に対応づけて示す情報であ  
る

ことを特徴とする請求項 1 記載の光ディスク。

20 1 3. 前記光ディスクにはプレイリスト情報、動的シナリオが  
記録されており、

プレイリスト情報は、動画データにおける再生区間を示す情報  
を、再生順序に従って配列することにより再生経路を定義する情  
報であり、

25 動的シナリオは、1 つ以上の再生経路の再生手順を示すことに  
より、映像タイトルを定義する情報であり、

インターリーブユニットは、識別情報を有しており、

識別情報は、再生経路、再生区間、映像タイトル全体、映像タ  
イトルの章の何れかを、プログラム又はデータの生存区間として  
30 示す

ことを特徴とする請求項 1 記載の光ディスク。

14. インターリーブユニットはロケータを含んでおり。

ロケータは、ドライブ情報と、パス情報とを含み、

5     ドライブ情報は、プログラム又はデータを読み出すべき読出先のドライブを示し、

パス情報は、そのドライブの階層構造のうち、どの階層にプログラム又はデータが存在するかを示す

ことを特徴とする請求項 1 記載の光ディスク。

10

15. デジタルストリームを記録した光ディスクについての再生装置であって、

デジタルストリームを構成する  $n$  個のセグメントのうち  $i$  番目セグメントを、 $i$  番目セグメントに先行するインターリーブユニットと共に光ディスクから読み出す読出手段と ( $i, n$  は、 $i < n$  を満たす整数)、

15

読み出された  $i$  番目セグメントを再生する再生手段と、

読み出されたインターリーブユニット内のプログラム又はデータを用いて、セグメント再生と同期した処理を行う処理手段と

20

を備えることを特徴とする再生装置。

16. 前記読出手段により読み出されたインターリーブユニットを格納するメモリと、

読出手段により読み出されたセグメントを格納するトラックバッファとを備え、

25

前記再生手段は、トラックバッファを介してセグメントの供給を受け、

前記処理手段は、メモリを介してインターリーブユニットの供給を受ける

30

ことを特徴とする請求項 15 記載の再生装置。

17. 前記再生装置は、スイッチャを備え、

前記読出手段は、光ディスクにおけるセクタを読み取る度に、そのセクタのアドレスの通知を行い、

5     スイッチャは、読出手段から通知されたセクタアドレスが、インターリーブユニットの占有領域内であれば、セクタからの読取情報をメモリに書き込み、

10     読出手段から通知されたアドレスが、セグメントの占有領域内であれば、セクタからの読取情報を順次トラックバッファに書き込み、

前記メモリ上のインターリーブユニットは、スイッチャによりメモリに書き込まれた読取情報から構成され、

15     前記トラックバッファ上のセグメントは、スイッチャによりトラックバッファに書き込まれた読取情報から構成されることを特徴とする請求項16記載の再生装置。

18. 前記インターリーブユニットと、デジタルストリームとは光ディスクにおいて別々のファイルに格納されており、

光ディスクにはファイル管理情報が記録されており、

20     ファイル管理情報は、ファイルの識別情報と、光ディスクにおけるデジタルストリーム又はインターリーブユニットのアドレスとを対応づけて示し、

25     現在の読取位置が光ディスクにおけるインターリーブユニットの占有領域内であるか、セグメントの占有領域内であるかのスイッチャによる判定は、ファイル管理情報を参照してなされることを特徴とする請求項17記載の再生装置。

19. 前記インターリーブユニットと、デジタルストリームとは光ディスクにおいて別々のファイルに格納されており、

30     光ディスクにはファイル管理情報が記録されており、



ファイル管理情報は、ファイルの識別情報と、光ディスクにおけるデジタルストリーム又はインターリーブユニットのアドレスとを対応づけて示し、

読出手段は、前記複数ファイルをオープンして、インターリーブユニットを格納したファイルの読出先をメモリに指定し、デジタルストリームを格納したファイルの読出先をトラックバッファに指定した上で読み出しを行う

ことを特徴とする請求項 16 記載の再生装置。

20 20. 前記インターリーブユニットは終了時点情報を含み、  
前記処理手段は、

再生手段による現在の再生時点が終了時点情報に示される終了時点になれば、メモリからインターリーブユニットを削除することを特徴とする請求項 16 記載の再生装置。

15

21. 前記処理手段は仮想マシン部を備え、

処理手段は、アプリケーションプログラムからの要求に応じて、前記メモリ上のインターリーブユニット内のプログラム又はデータを仮想マシン部内のワーク領域に供給して、仮想マシン部に

20 実行させる

ことを特徴とする請求項 16 記載の再生装置。

22. 前記インターリーブユニットは開始時点情報を含み、

処理手段が、アプリケーションプログラムからの要求に応じて、  
25 前記メモリ上のインターリーブユニット内のプログラムを仮想マシン部内のワーク領域に供給するのは、再生手段による現在の再生時点が開始時点情報に示される開始時点に達した場合であり、

再生手段による現在の再生時点が開始時点情報に示される開始時点に達していなければ、アプリケーションプログラムからの  
30

要求があってもインターリーブユニット内のプログラムを仮想マシン部内のワーク領域に供給しない、請求項 21 記載の再生装置。

5        23. 前記第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの境界は、プログラム又はデータの生存区間の途中にあたる位置であり、

第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの間には、前記インターリーブユニットの複製物が記録されており、

10        前記読出手段は、

$i$  番目セグメントの内部に頭出しを行う場合、前記複製物を光ディスクから読み出す、請求項 15 記載の再生装置。

15        24. 前記第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの境界は、前記プログラムの生存区間の終了点より後にあり、

第  $i$  番目のセグメントと、第  $i+1$  番目のセグメントとの間にインターリーブユニットの複製物が、記録されており、

前記読出手段が  $i$  番目セグメントを読み出すのは、再生手段が通常再生を行う場合であり、

20        再生手段が逆方向再生を行う場合、デジタルストリームを構成する複数セグメントのうち  $i+1$  番目セグメントを、 $i+1$  番目セグメントに先行する複製物と共に光ディスクから読み出すことを特徴とする請求項 15 記載の再生装置。

25        25. 前記  $i$  番目のセグメントは、複数のアクセスユニットからなり、各アクセスユニットはイントラ符号化されたピクチャを有した動画データを含み、

光ディスクにはマップ情報が記録されており、

30        マップ情報は、 $i$  番目セグメントに先行するインターリーブユニットを、第  $i$  番目セグメントに属するアクセスユニットのアド

レス又は再生時間と対応づけて示しており、

前記読出手段によるインターリーブユニットの読み出しは、マップ情報を参照してなされることを特徴とする請求項 15 記載の再生装置。

5

26. 前記処理手段は、仮想マシン部を備え、

再生手段は、デジタルストリームを再生すると共に、再生に同期したイベントの発生を行い、

10 前記仮想マシン部は、再生手段がイベントを発生すると、インターリーブユニット内のプログラムを実行することを特徴とする請求項 15 記載の再生装置。

27. 前記イベントとは、

15 現在の再生位置が、動画データの再生時間軸における所定の時点に到達したことを示すイベント、

再生装置が発生するイベント、

他のプログラムにより発生されるイベントの何れかである、ことを特徴とする請求項 26 記載の再生装置。

20 28. 前記光ディスクにはマーク情報が記録されており、

マーク情報は、

前記再生時間軸における所定の時点、及び、所定の時間間隔を定義する情報であり、

25 再生手段によるイベント発生は、このマーク情報に基づいて行われる

ことを特徴とする請求項 27 記載の再生装置。

29. 前記再生装置は、ユーザ操作を受け付ける受付手段を備え、

30 前記イベントとは、

動画データの再生時間軸上の所定の時間帯において、受付手段がユーザ操作を受け付けたことを示すイベントである請求項 27 記載の再生装置。

- 5      30. 光ディスクには、インターリーブユニット統合情報が記録されており、インターリーブユニット統合情報は、光ディスクに記録された複数インターリーブユニットに対する管理情報であり、
- 前記再生装置は、メモリ、再生制御手段を備え、
- 10      再生制御手段は、複数インターリーブユニットを、前記メモリに格納することができるか否かを、インターリーブユニット統合情報に基づき判定し、
- 前記読出手段は、前記再生制御手段が読み出すことができると判定した場合に複数インターリーブユニットの一部又は全部の
- 15      読み出しを行う、請求項 15 記載の再生装置。

31. 前記光ディスクにはプレイリスト情報が記録されており、プレイリスト情報は、デジタルストリームにおける再生区間を示す情報を、再生順序に従って配列することにより再生経路を定義する情報であり、
- 20      前記再生制御手段は、プレイリスト情報及び動的シナリオを用いたデジタルストリームの再生を行うよう読出手段及び再生手段を制御する

ことを特徴とする請求項 15 記載の再生装置。

25

32. インターリーブユニットは、識別情報を有しており、前記再生制御手段は、プレイリスト情報を用いたデジタルストリームの再生を行うにあたって、

そのプレイリスト情報の識別情報を有したインターリーブユ

30      ニット、

そのプレイリスト情報における再生区間を示す情報の識別情報を有したインターリーブユニット、  
の何れかを光ディスクから読み出すよう読出手段を制御することを特徴とする請求項 3 1 記載の再生装置。

5

3 3. 前記光ディスクには動的シナリオが記録されており、  
動的シナリオは、プレイリスト情報に示される 1 つ以上の再生経路の再生手順を示すことにより、映像タイトルを定義する情報であり、

10

前記再生制御手段は、  
その動的シナリオに対応するタイトルの識別情報を有したインターリーブユニット、  
その動的シナリオに対応するタイトル中のチャプターの識別情報を有したインターリーブユニット、

15

の何れかを光ディスクから読み出すよう読出手段を制御することを特徴とする請求項 3 1 記載の再生装置。

3 4. インターリーブユニットはロケータを含んでおり。

ロケータは、ドライブ情報と、パス情報とを含み、

20 ドライブ情報は、プログラム又はデータを読み出すべき読出先のドライブを示し、

パス情報は、そのドライブの階層構造のうち、この階層にプログラム又はデータが配すべきかを示し、

25 前記再生装置は、ロケータのドライブ情報に示されるドライブの階層構造のうち、パス情報に示される階層に、プログラム又はデータを配する

ことを特徴とする請求項 1 5 記載の再生装置。

3 5. デジタルストリームを記録した光ディスクについての再生プログラムであって、

30

デジタルストリームを構成する複数セグメントのうち  $i$  番目セグメントを、 $i$  番目セグメントに先行するインターリーブユニットと共に光ディスクから読み出す読出ステップと、

読み出された  $i$  番目セグメントを再生する再生ステップと、

- 5   読み出されたインターリーブユニット内のプログラム又はデータを用いて、セグメント再生と同期した処理を行う処理ステップと

をコンピュータに実行させることを特徴とするプログラム。

- 10   36. デジタルストリームを記録した光ディスクについての再生方法であって、

デジタルストリームを構成する複数セグメントのうち  $i$  番目セグメントを、 $i$  番目セグメントに先行するインターリーブユニットと共に光ディスクから読み出す読出ステップと、

- 15   読み出された  $i$  番目セグメントを再生する再生ステップと、

読み出されたインターリーブユニット内のプログラム又はデータを用いて、セグメント再生と同期した処理を行う処理ステップと

を有する再生方法。

20

37. 光ディスクの記録方法であって、

アプリケーションデータを作成するステップと、

作成したデータを光ディスクに記録するステップとを有し、

前記アプリケーションデータは、デジタルストリームと、イン

- 25   ターリーブユニットとを含み、

デジタルストリームは、 $n$  個のセグメントに分割されて光ディスクに記録されており、

$n$  個のセグメントのうち、 $i$  番目に再生されるものの前に、インターリーブユニットが記録され、

- 30   ここで  $i, n$  は、 $i < n$  を満たす整数であり、

インターリーブユニットは、 $i$  番目のセグメントの再生に同期するプログラム、又は、  
 $i$  番目のセグメントの再生に同期して表示されるデータを含む。  
ことを特徴とする記録方法。

図1

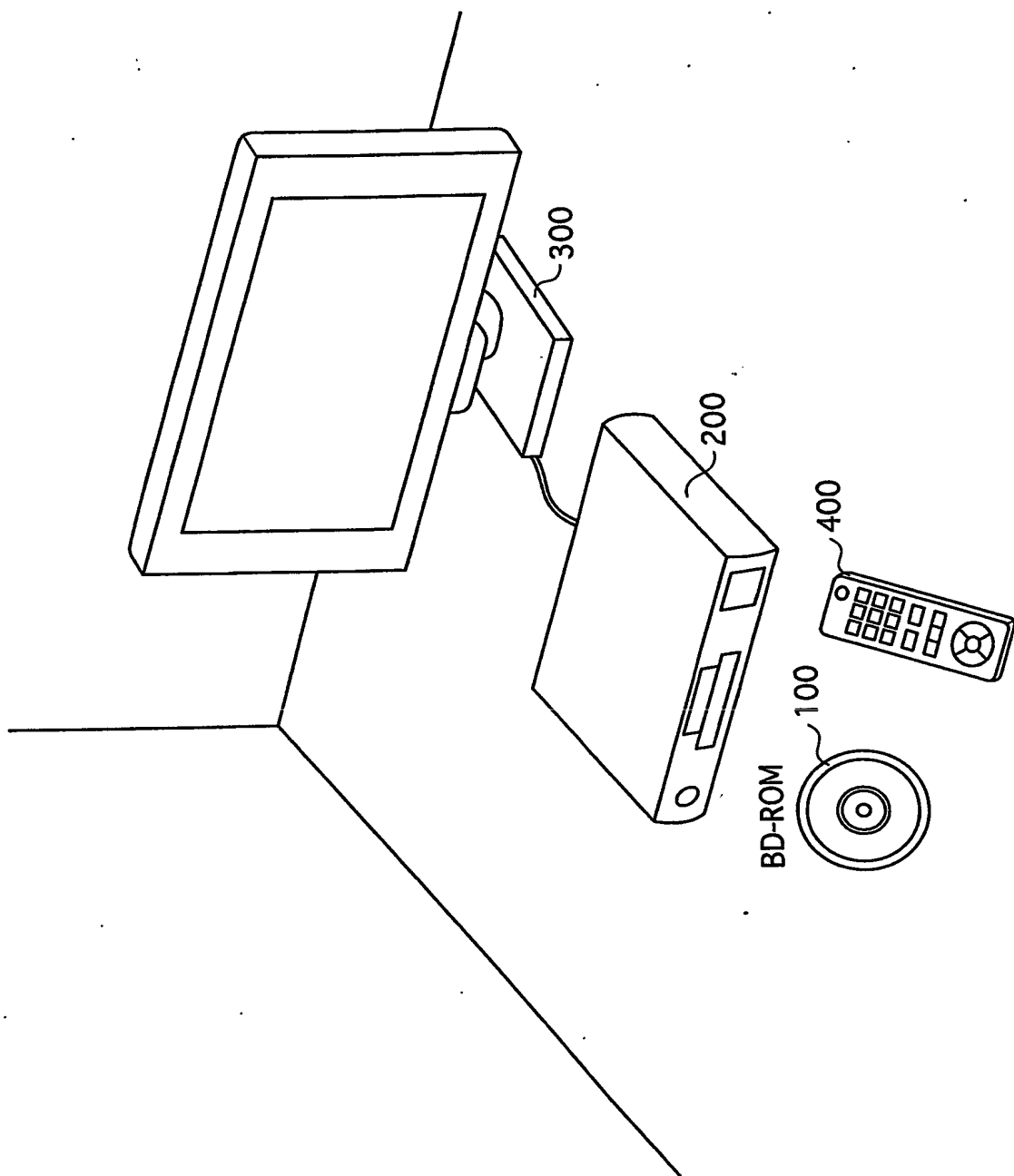




図2

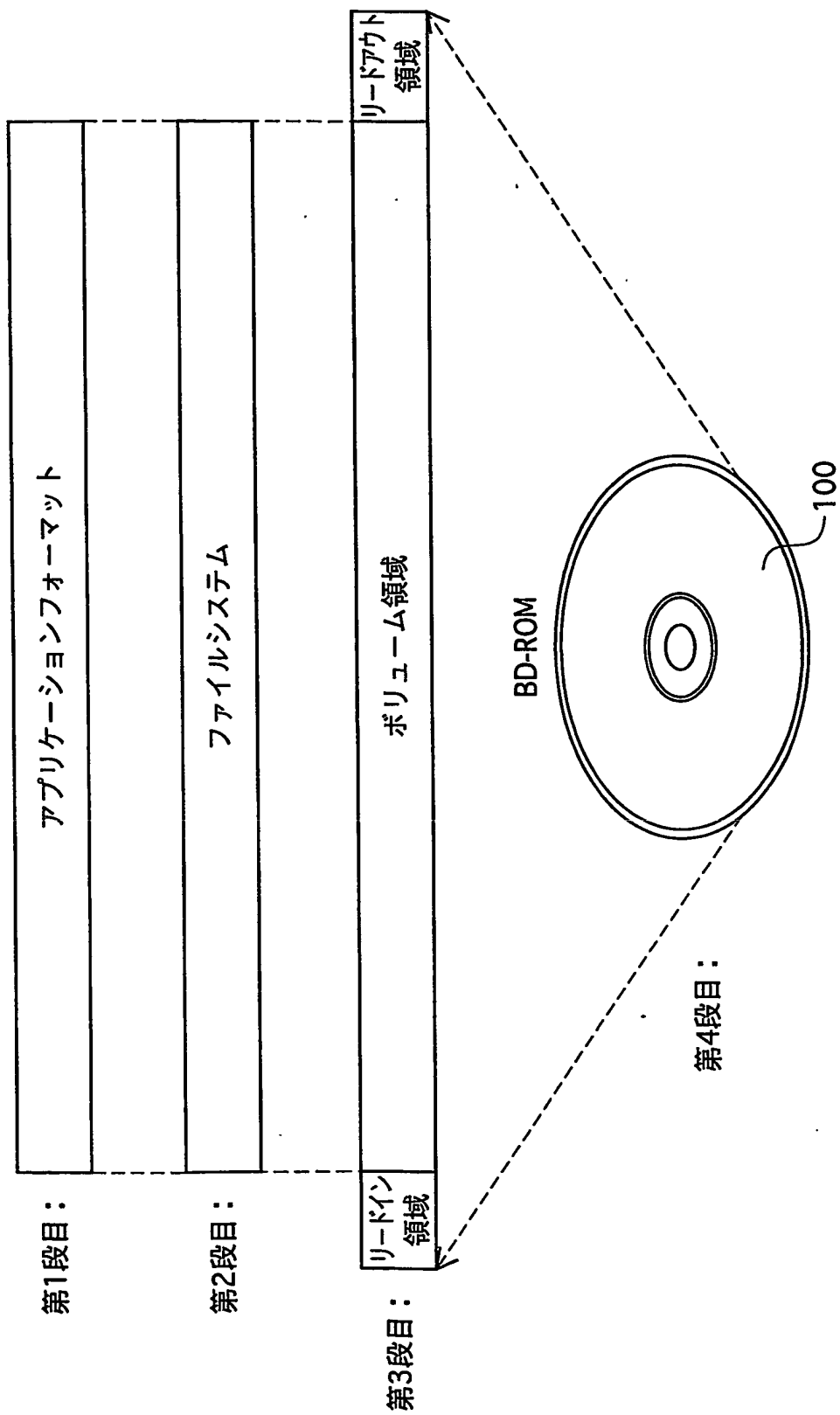


図3

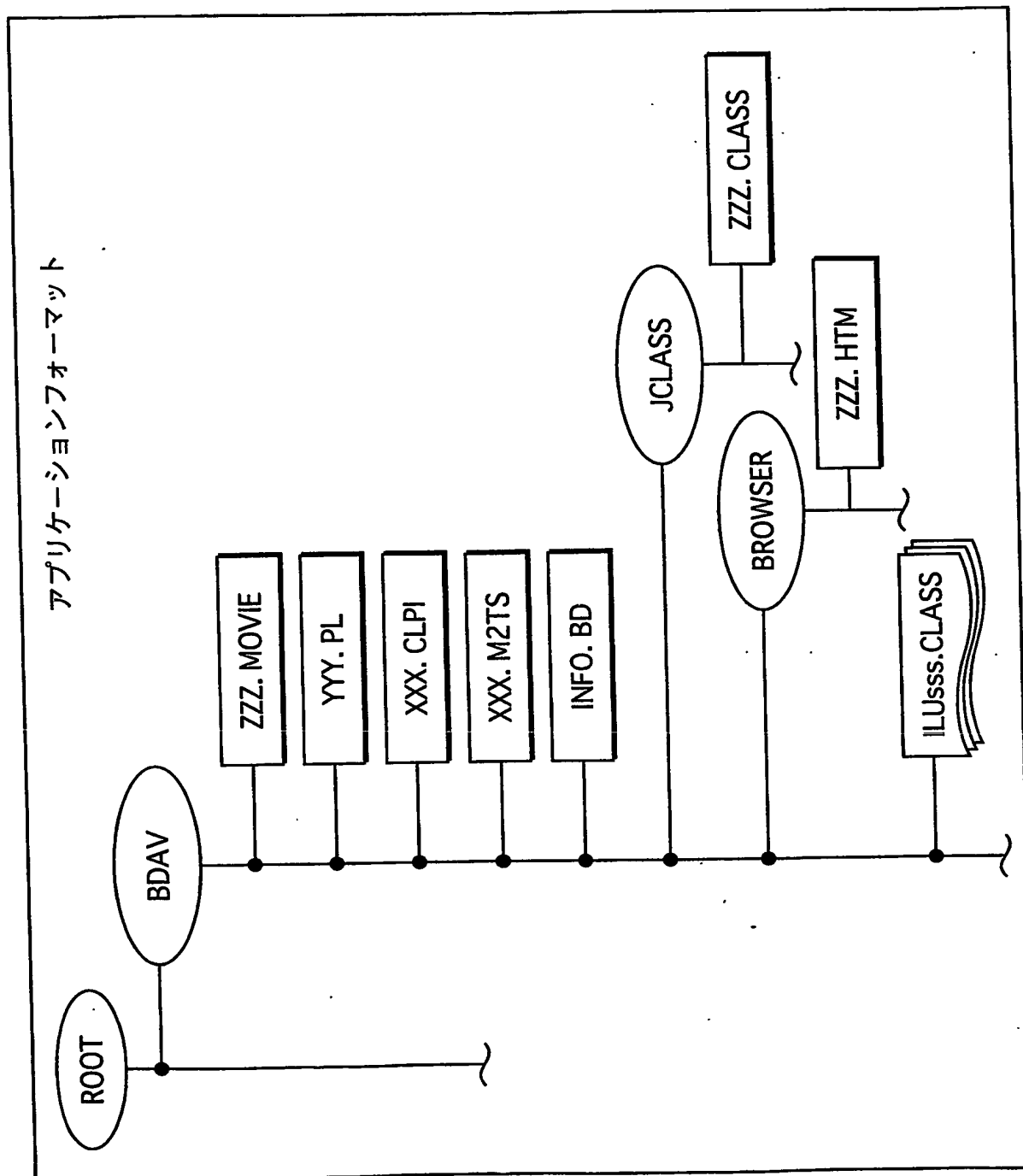


図4

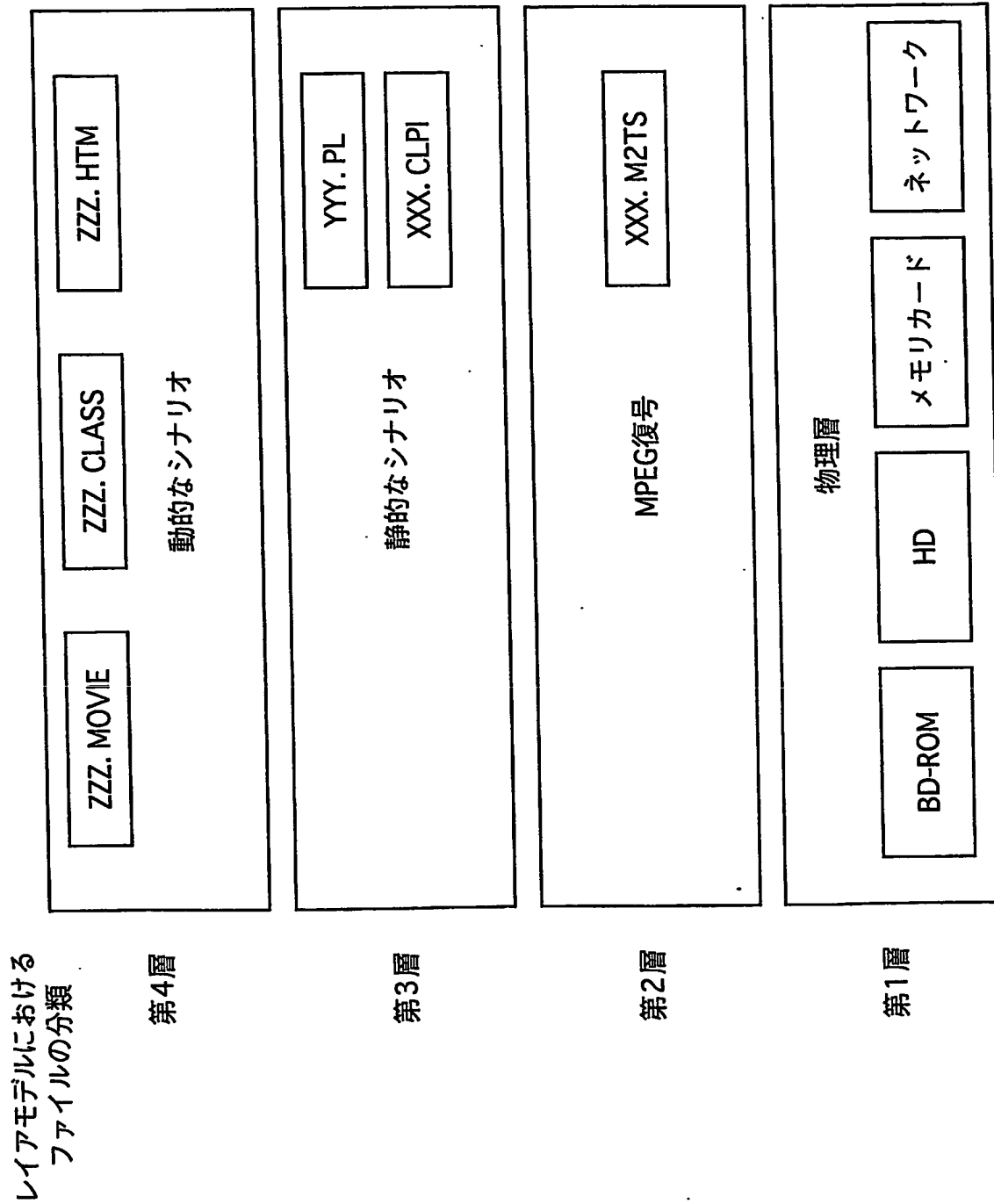


図5

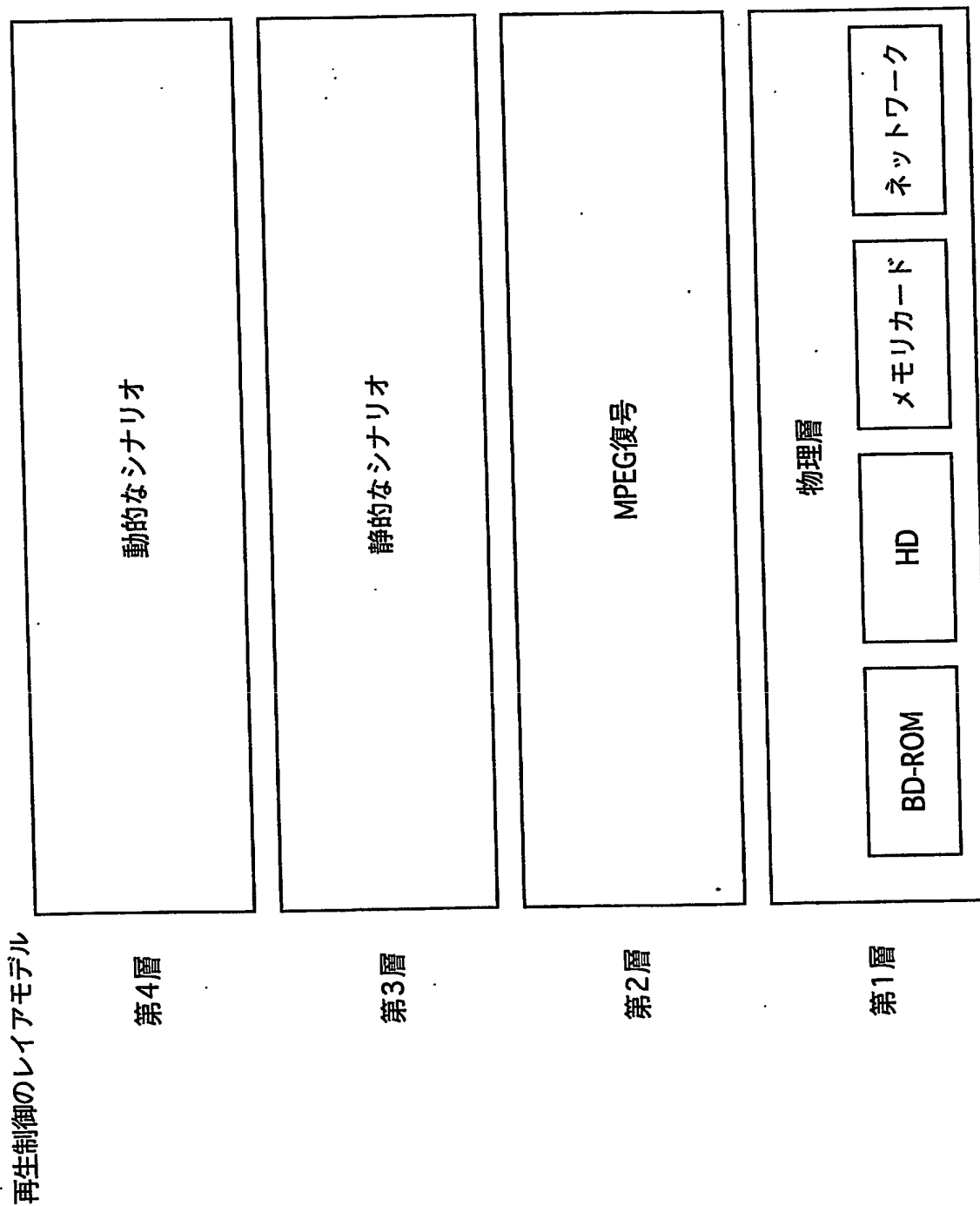


図6

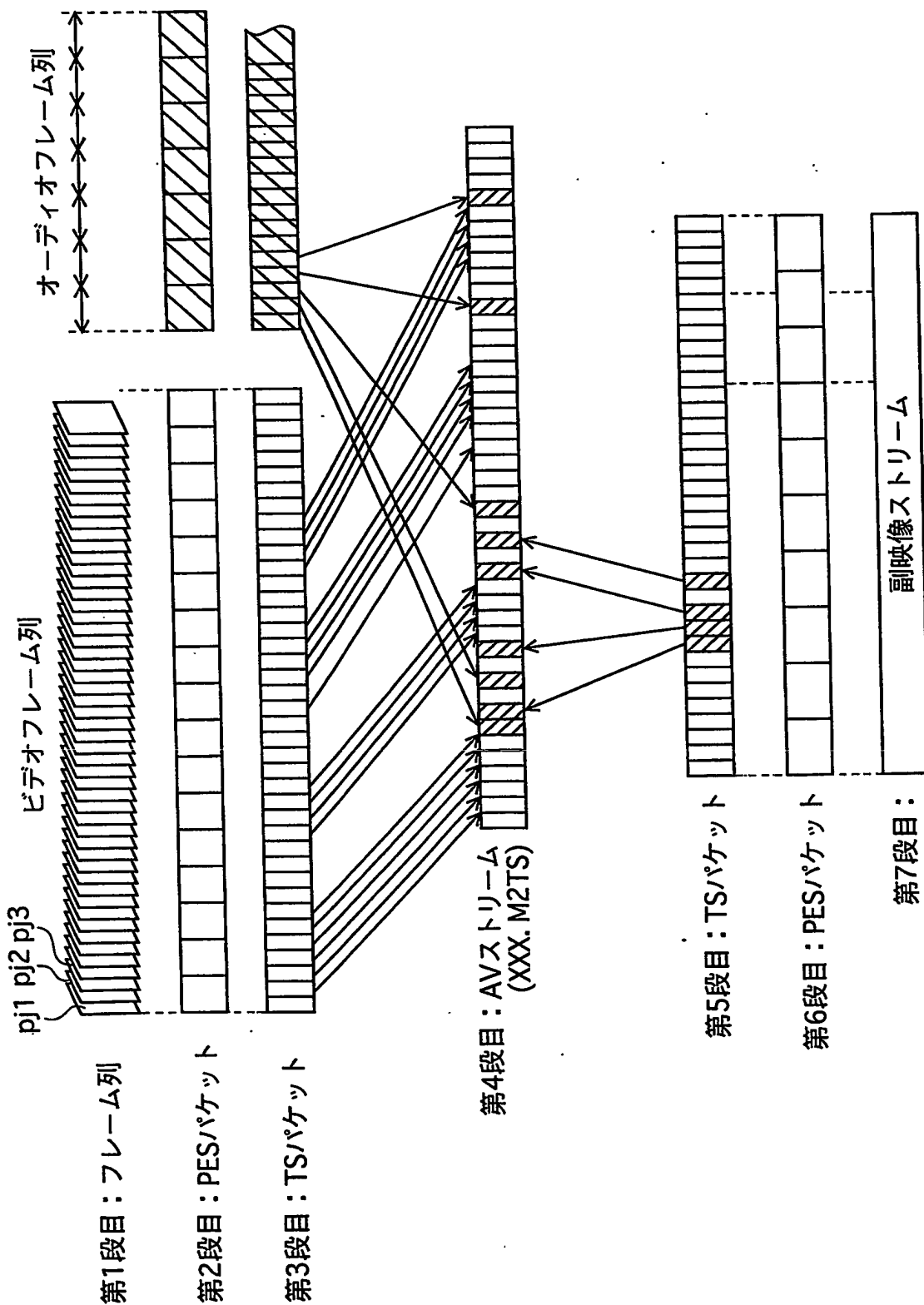


図7

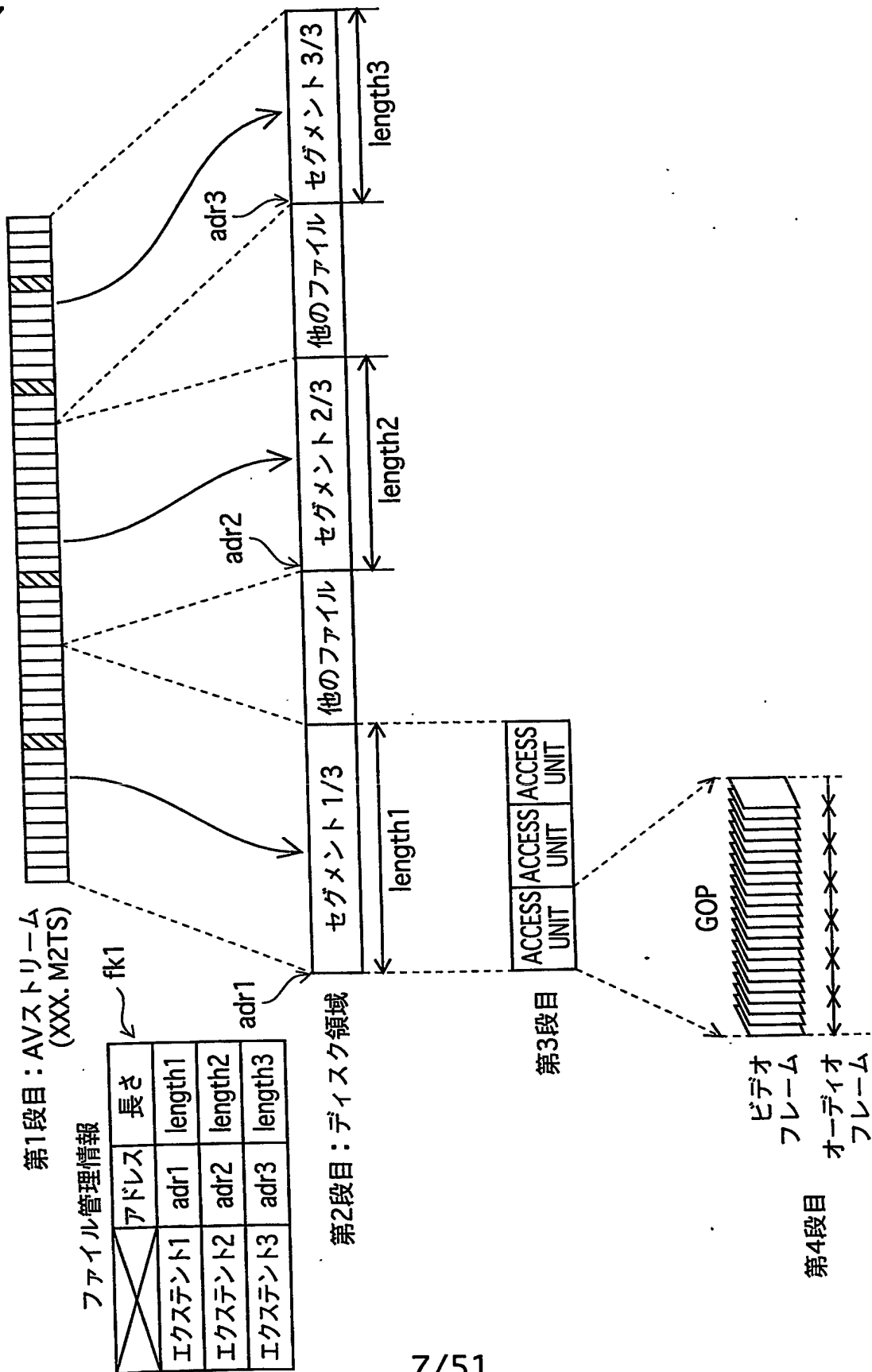


図8

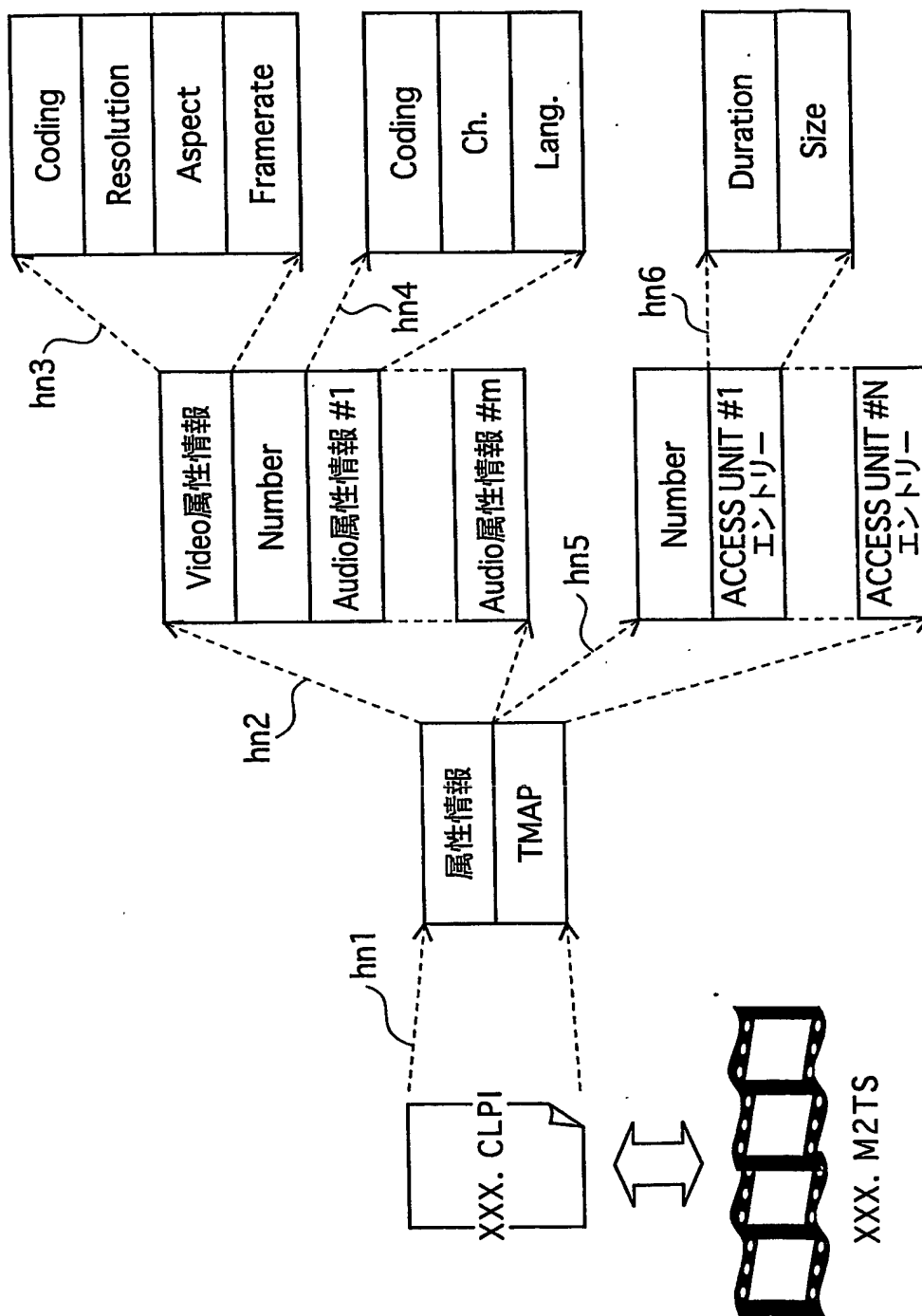


図9

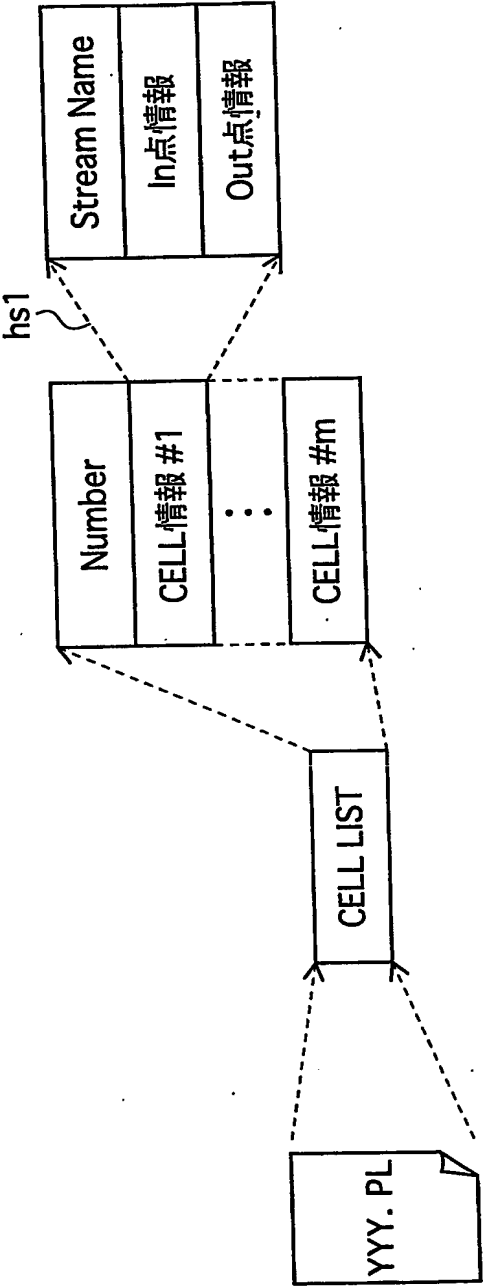




図10

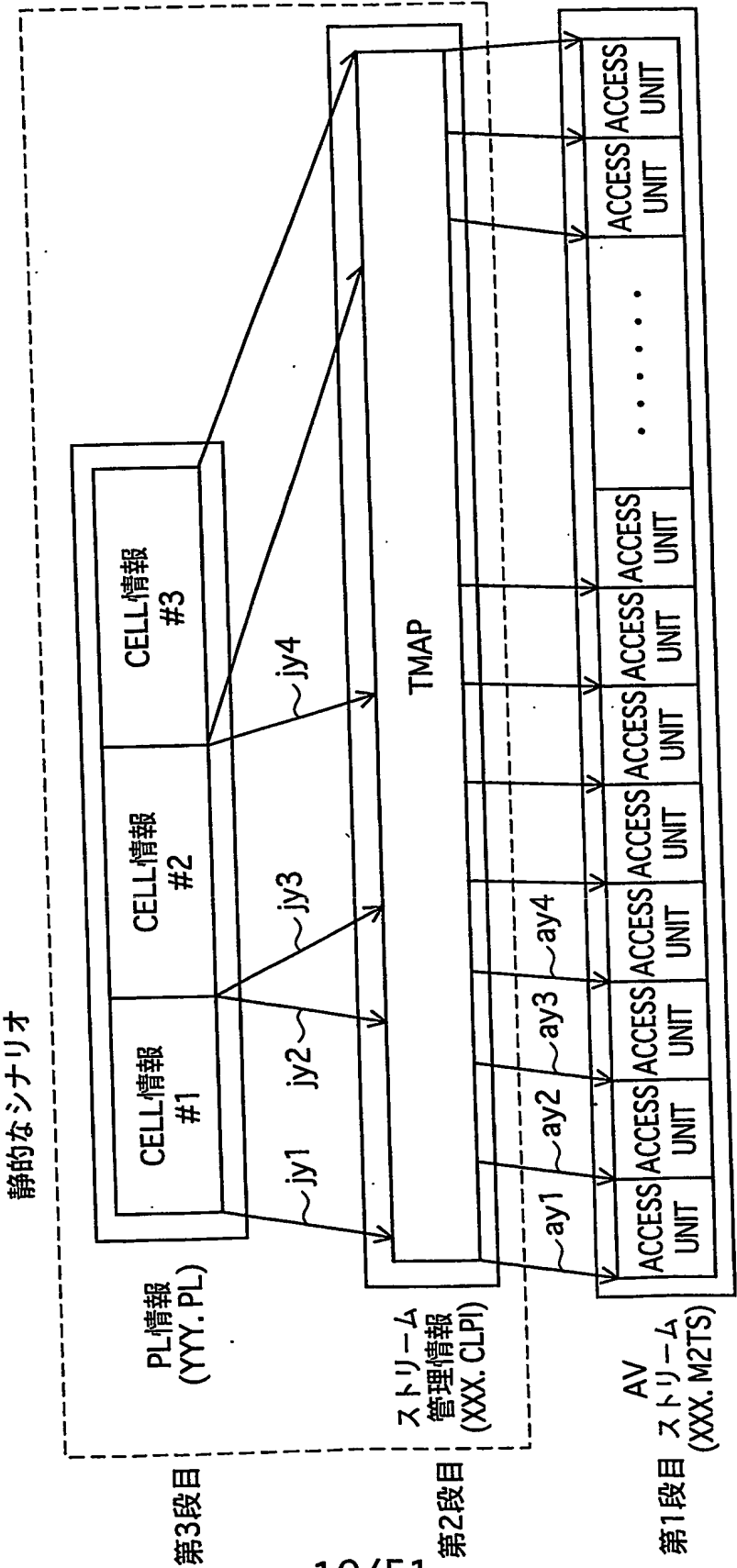


圖 11

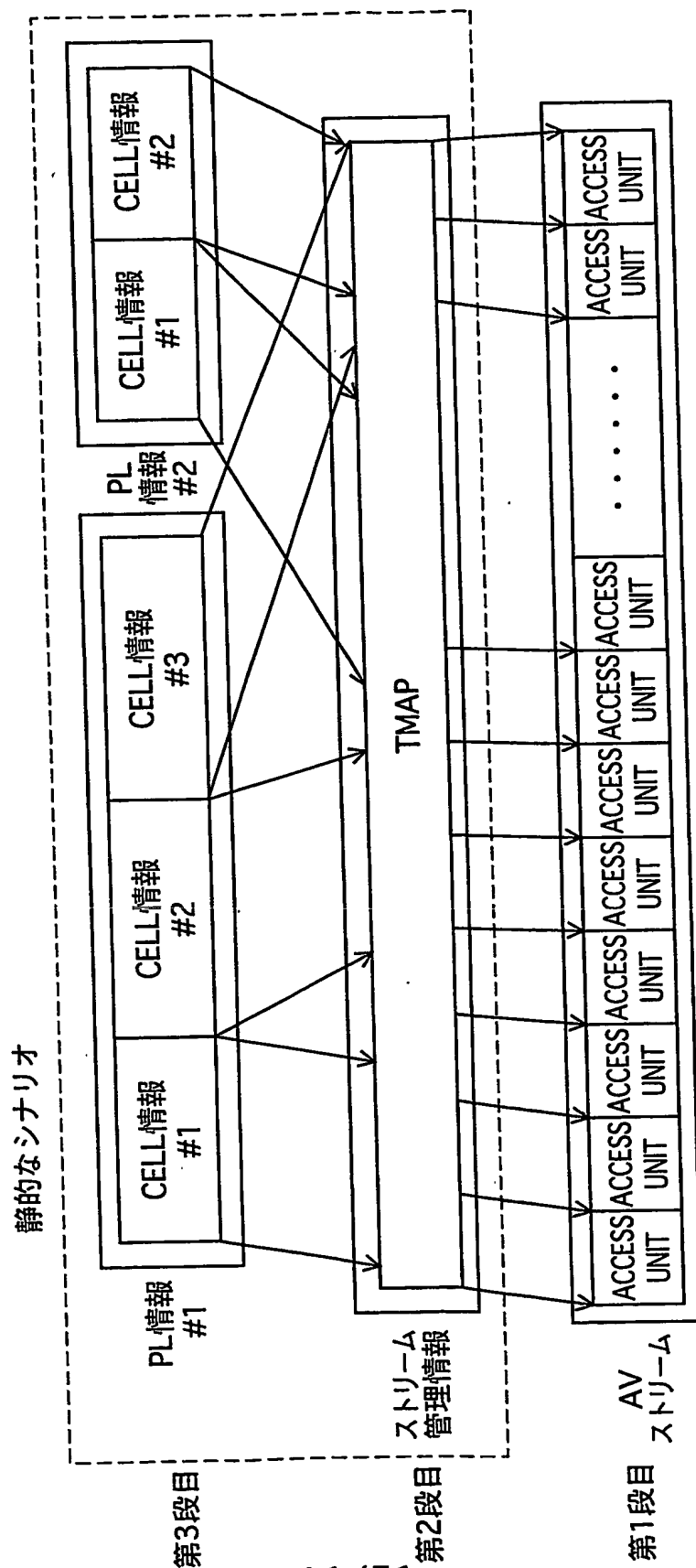


図12

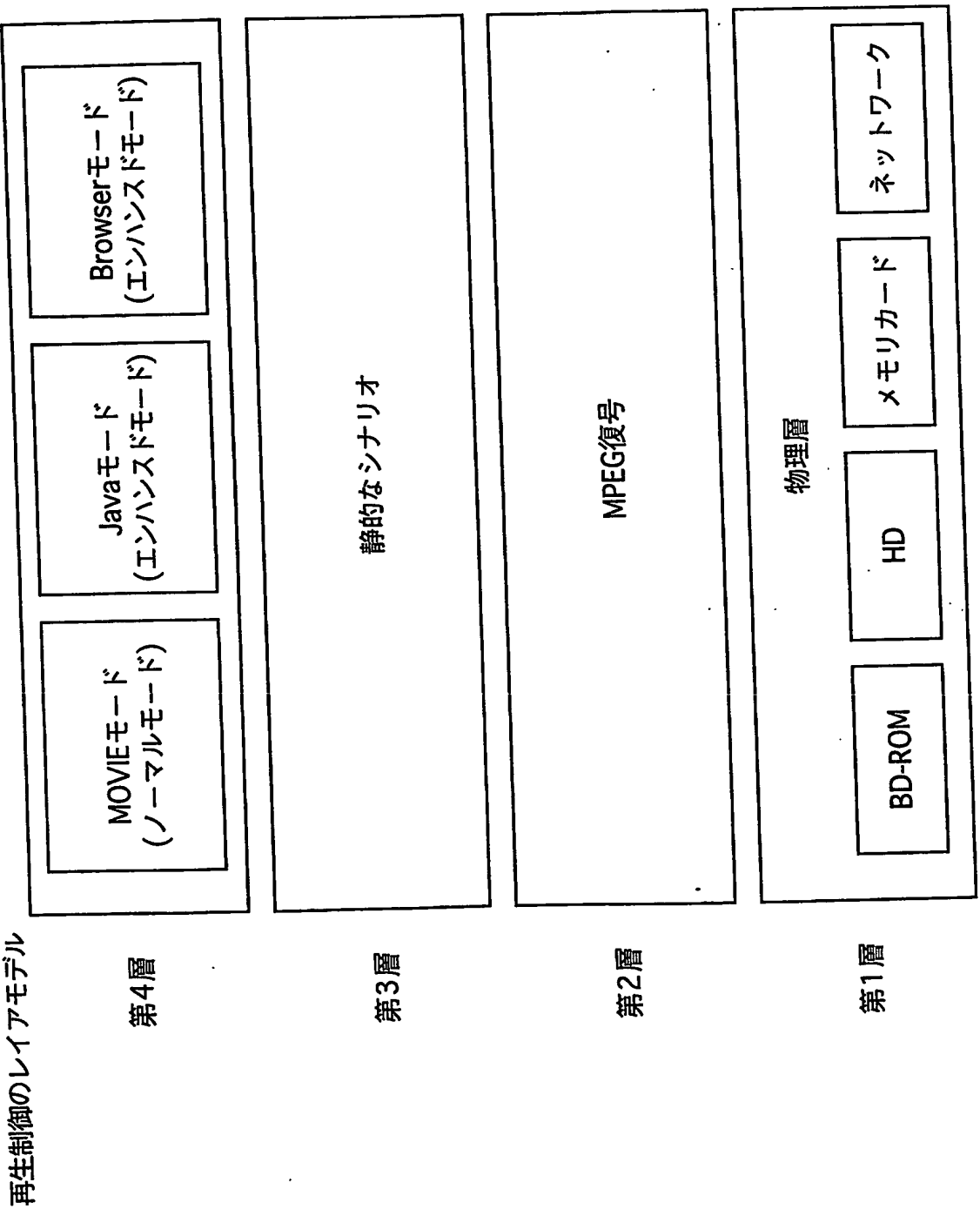


図13

Java 言語が対象とする制御ソフトウェアのレイアモデル

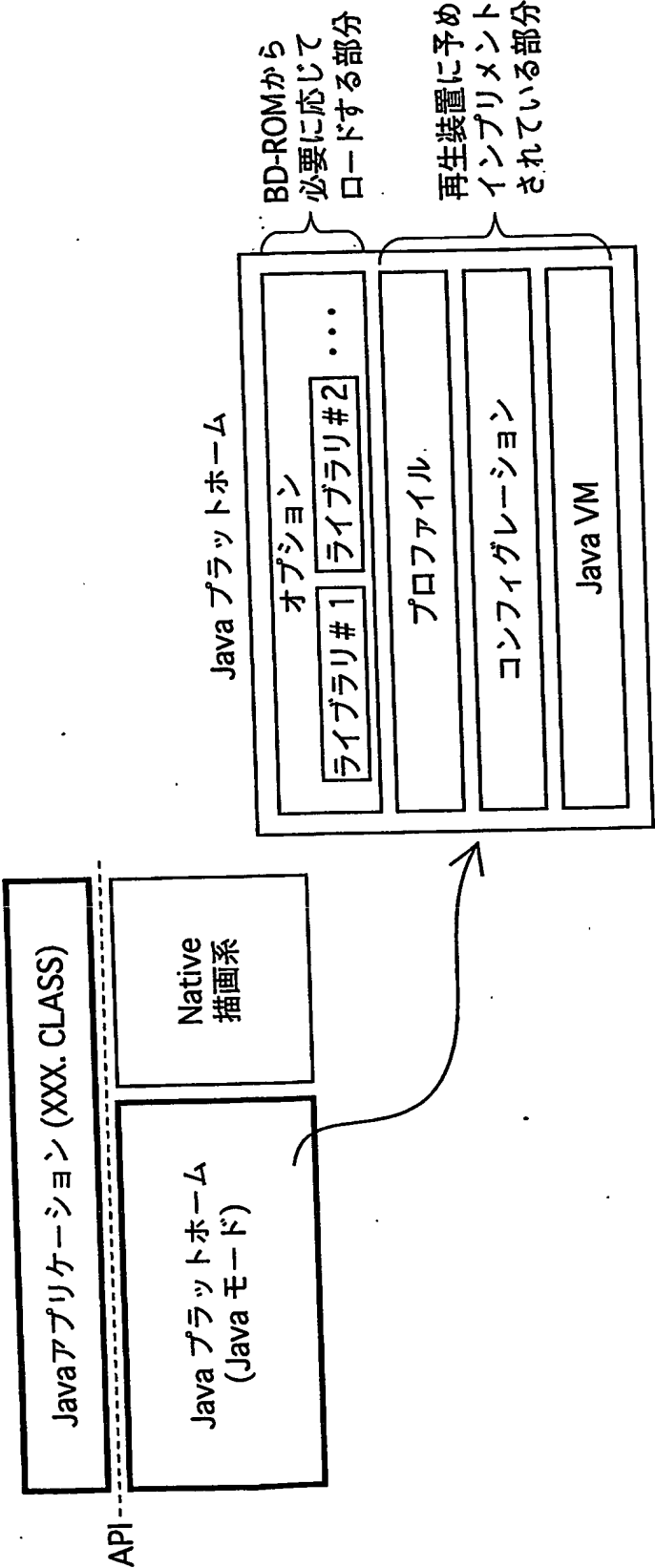


図14

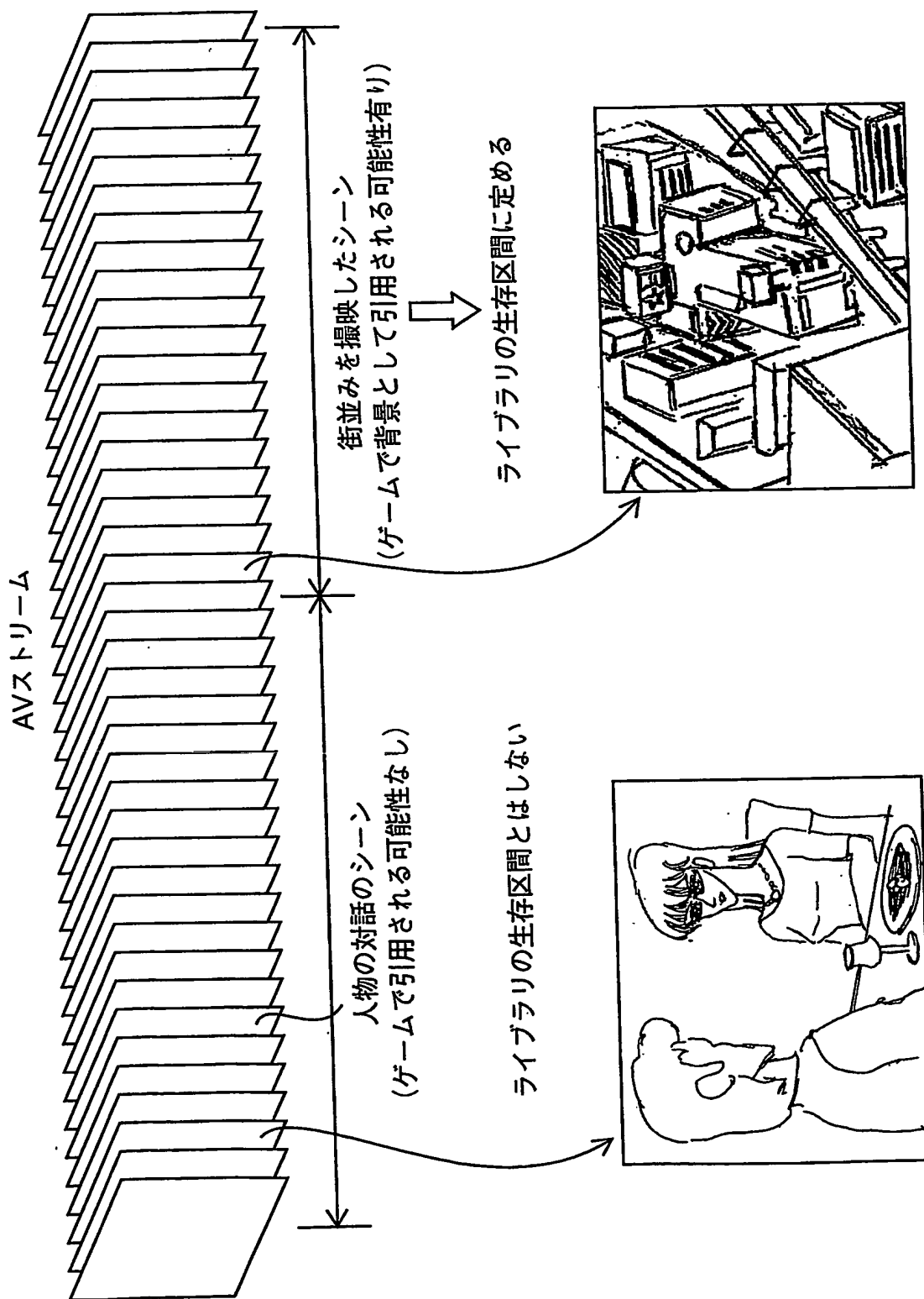
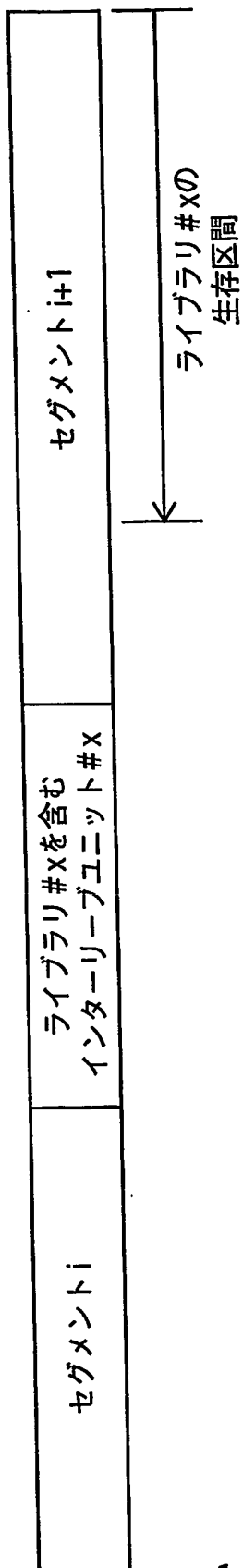


図15

(a)



(b)

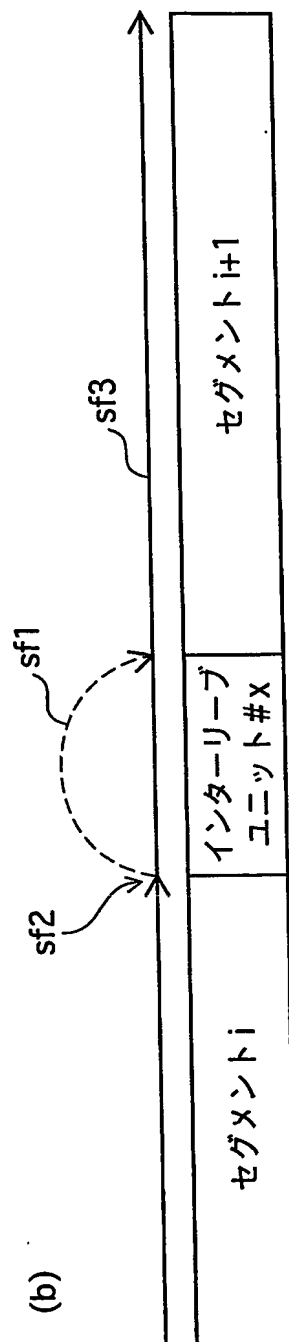


図16

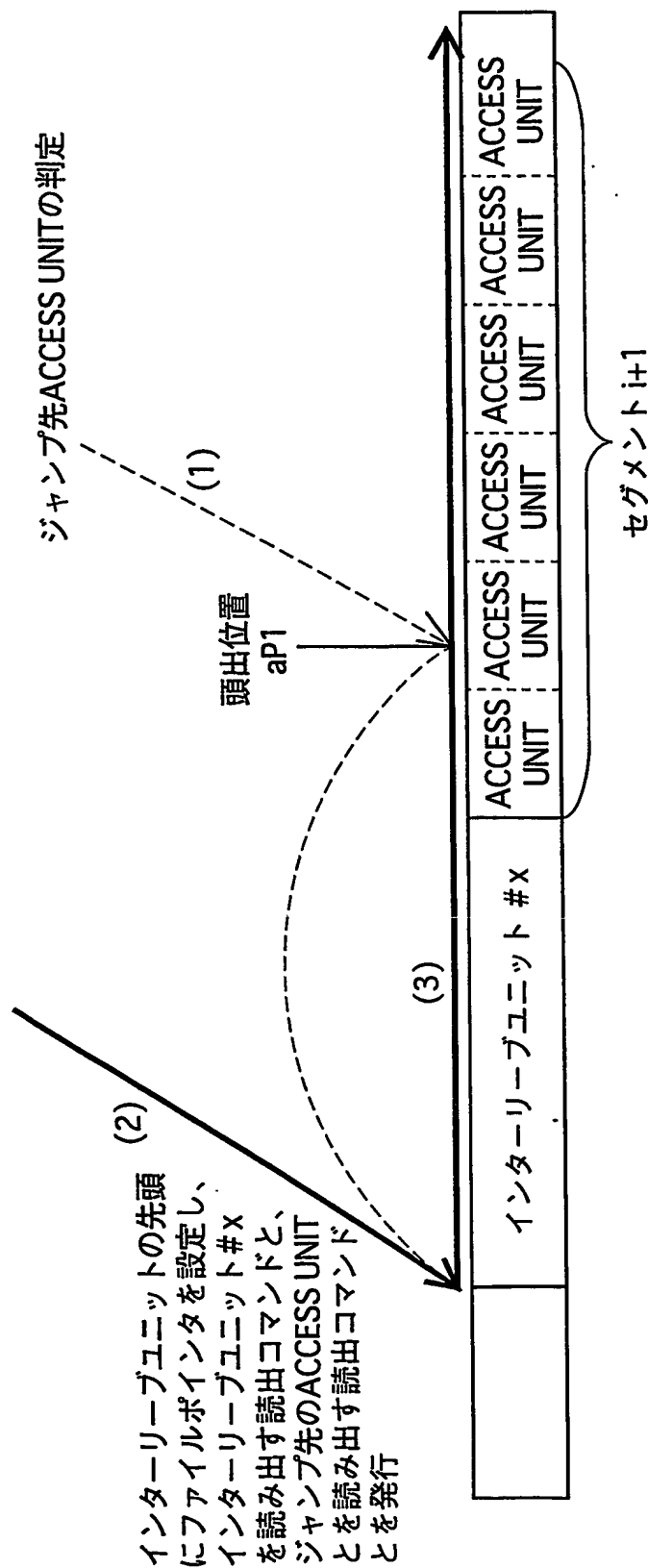


図17

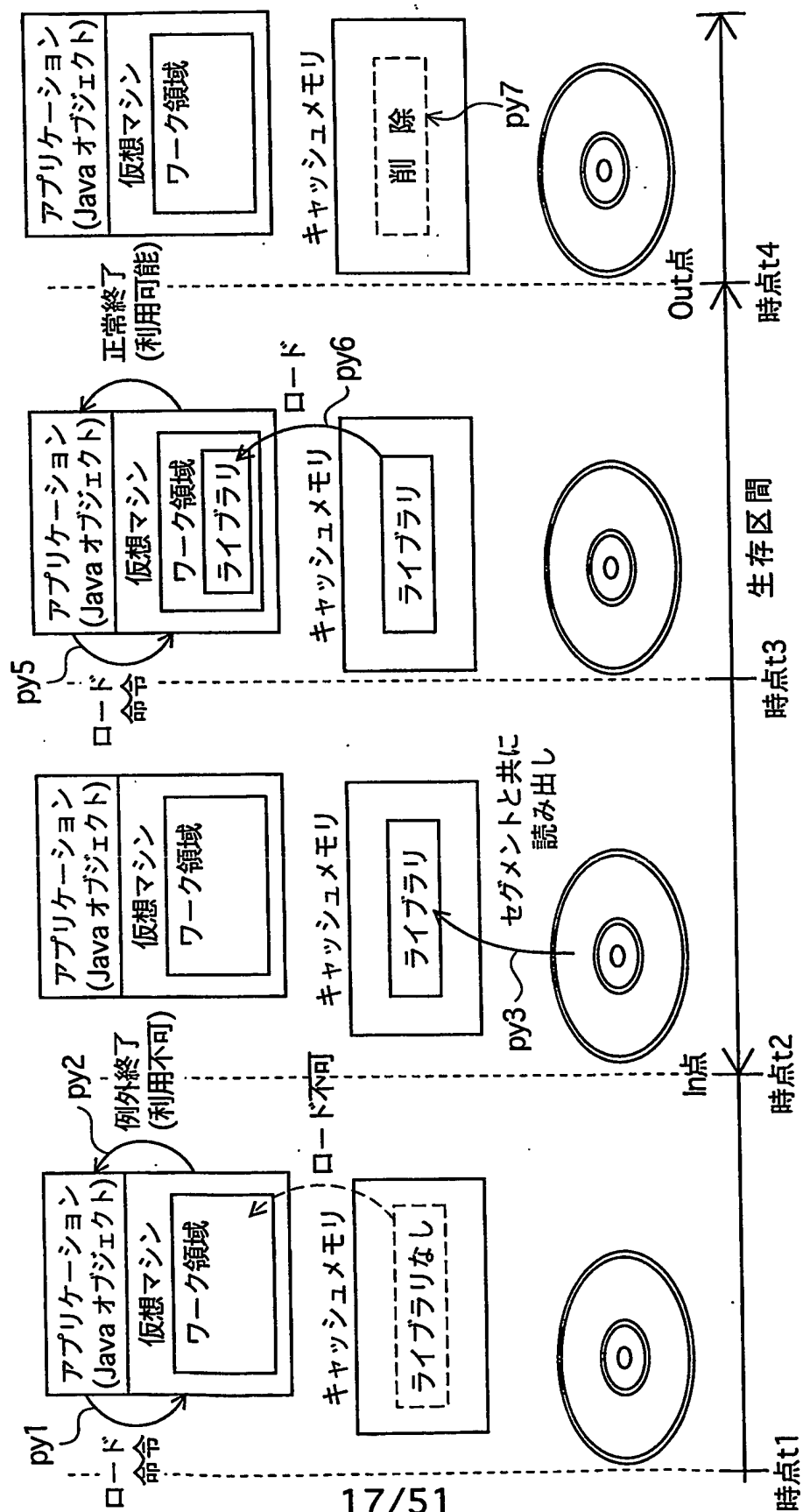




図18

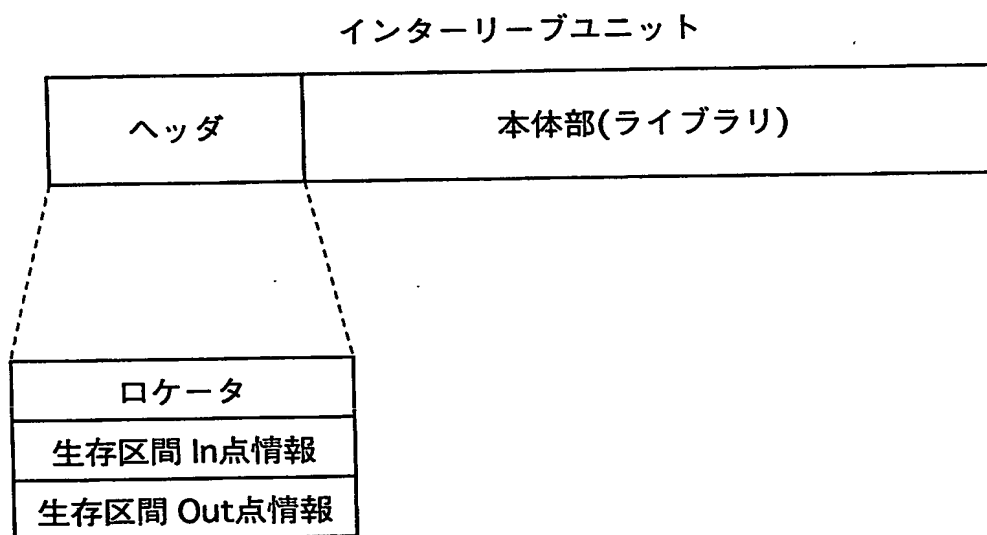


図19

タイムマップ

	Duration	Size	先行するインターリーブユニット
ACCESS UNIT#1	0	0	ILU001
ACCESS UNIT#20	135,000	8,000	ILU002
ACCESS UNIT#30	270,000	16,000	ILU003

図20

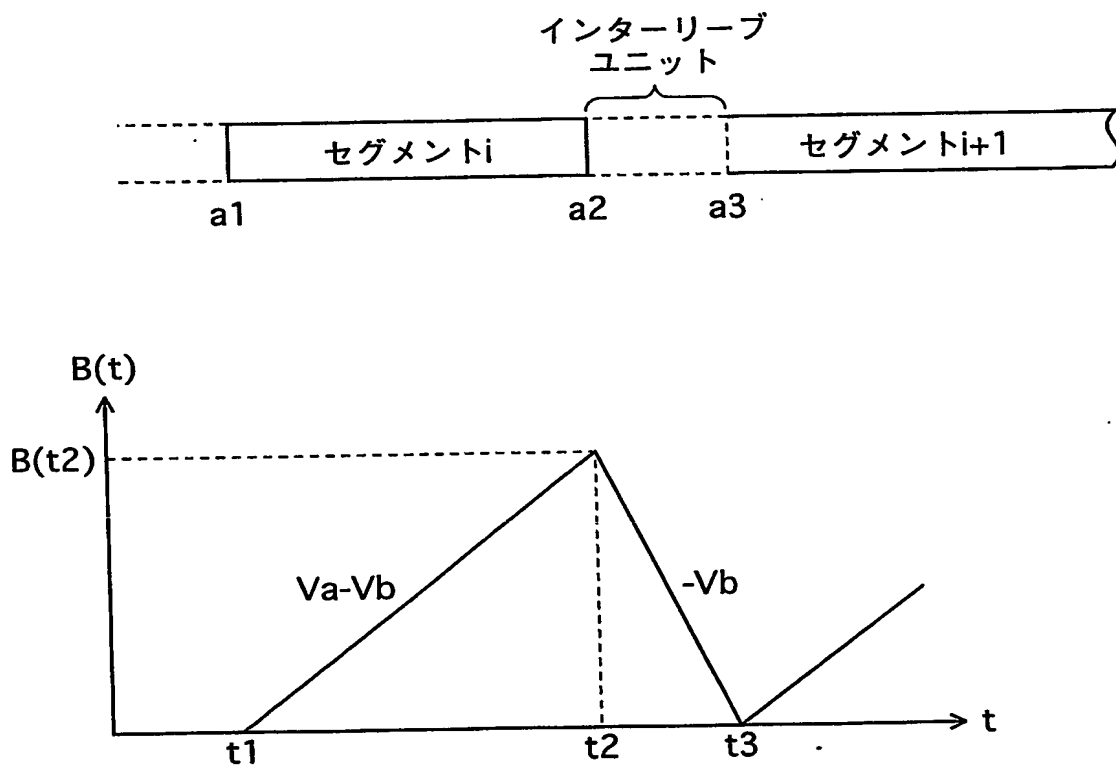


図21

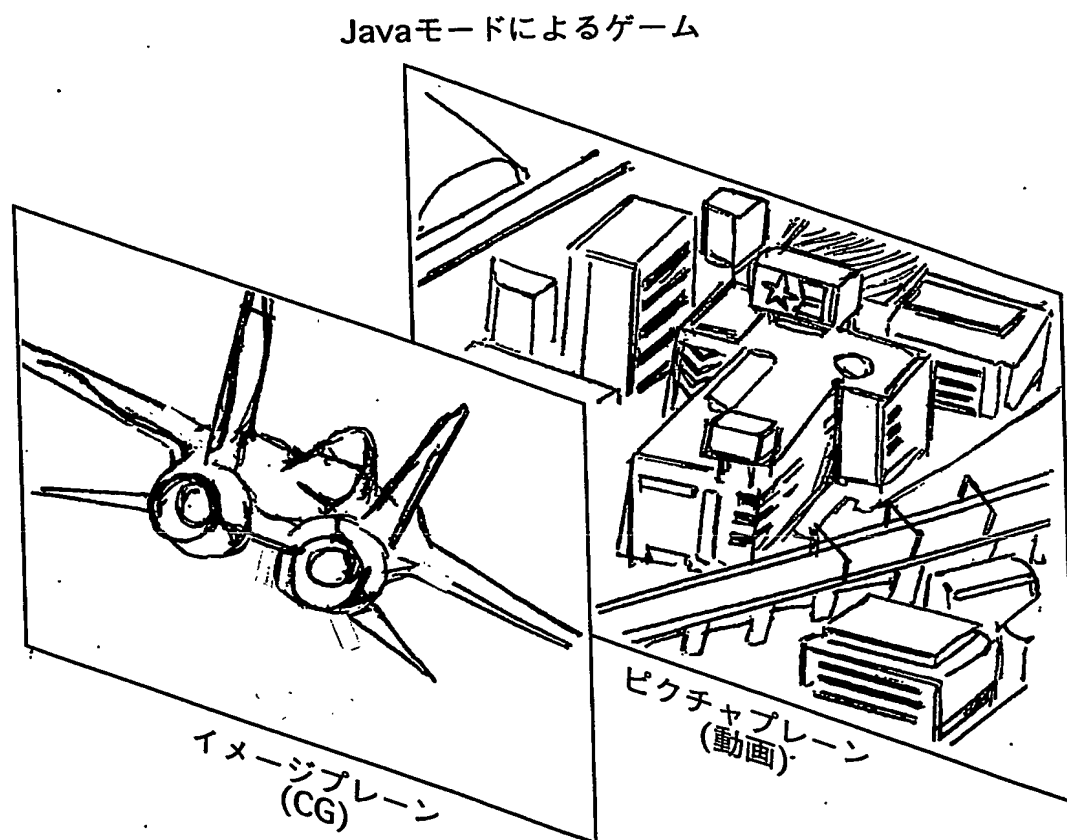


図22

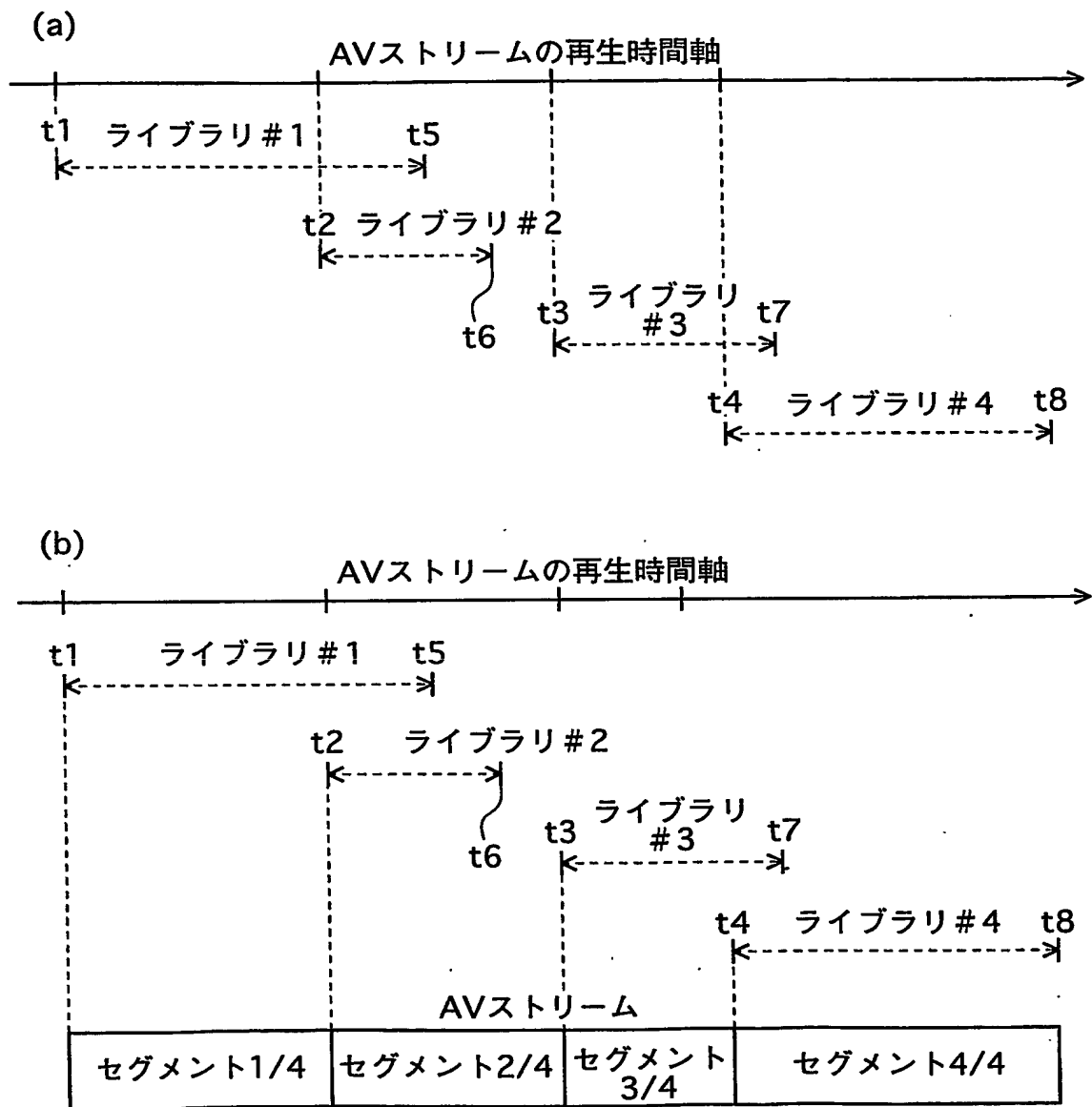


図23

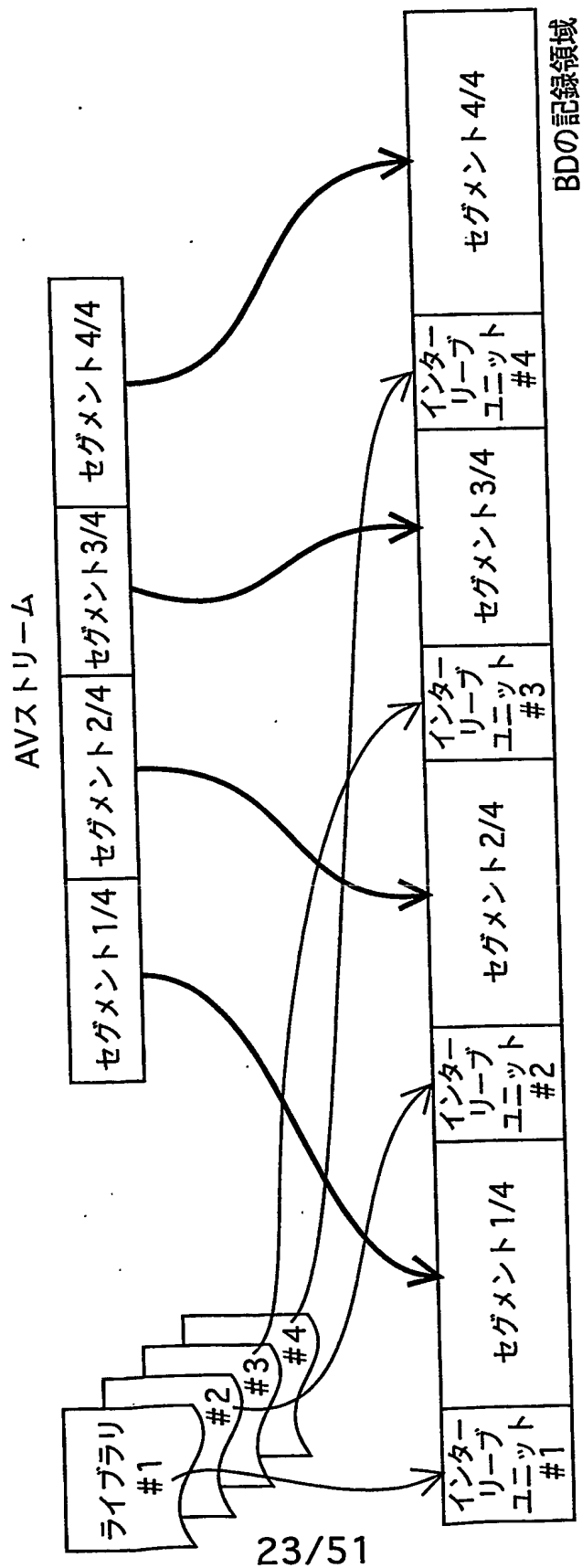


図24

## インターリーブユニット # 1

生存区間In点 . . . . t1 生存区間Out点 . . . . t5	ライブラリ # 1
---	-----------

## インターリーブユニット # 2

生存区間In点 . . . . t2 生存区間Out点 . . . . t6	ライブラリ # 2
---	-----------

## インターリーブユニット # 3

生存区間In点 . . . . t3 生存区間Out点 . . . . t7	ライブラリ # 3
---	-----------

## インターリーブユニット # 4

生存区間In点 . . . . t4 生存区間Out点 . . . . t8	ライブラリ # 4
---	-----------

図25

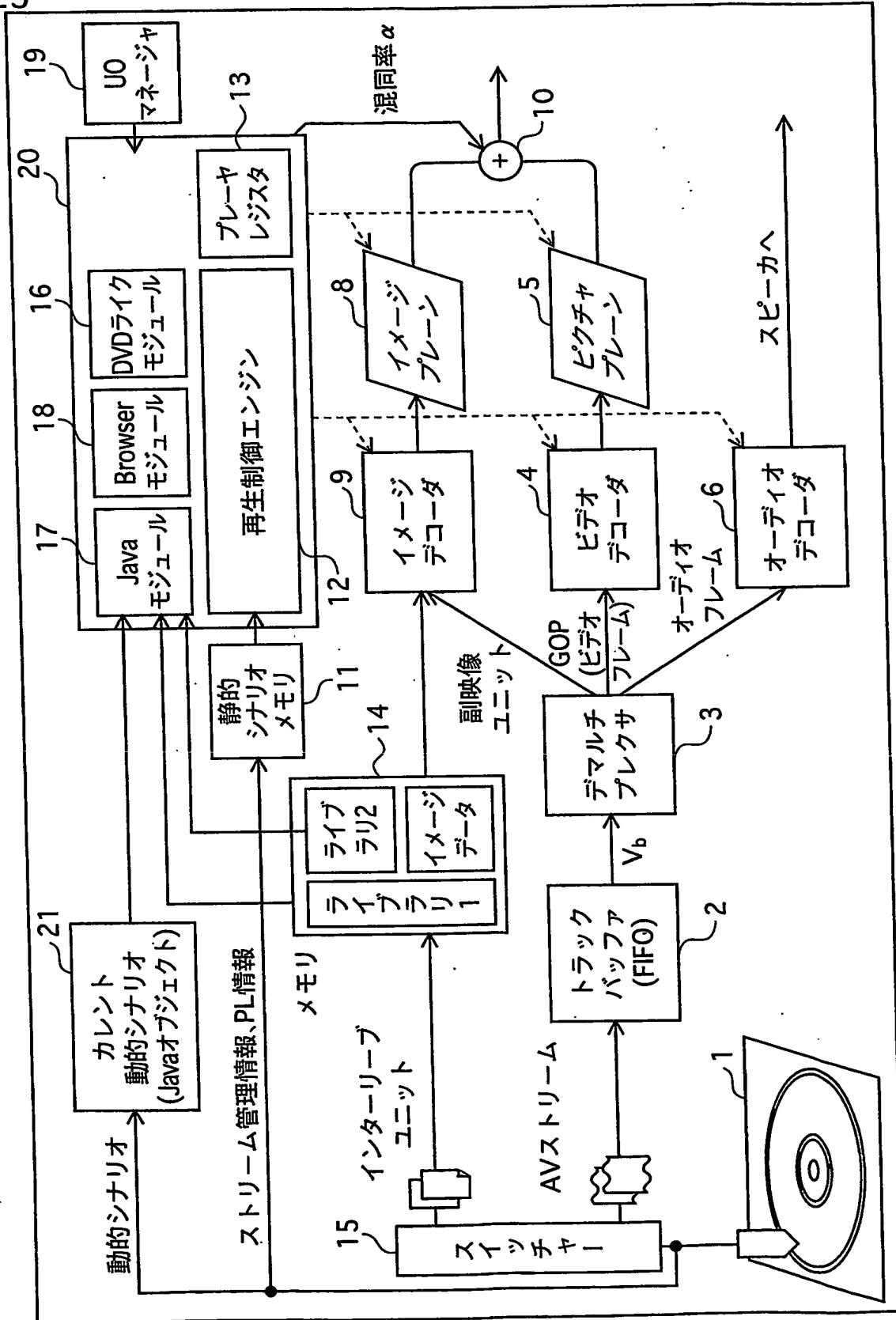




図26

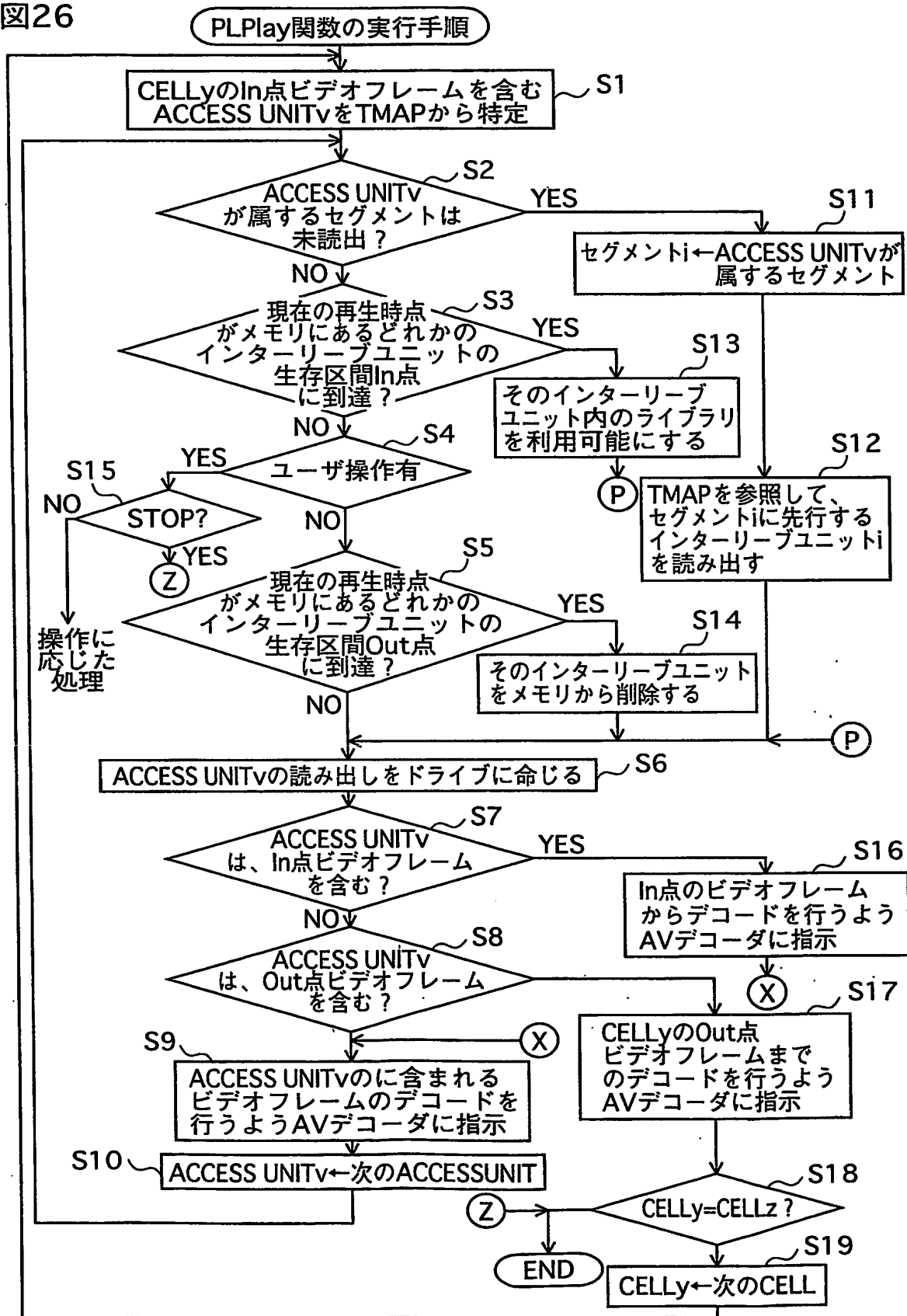


図27

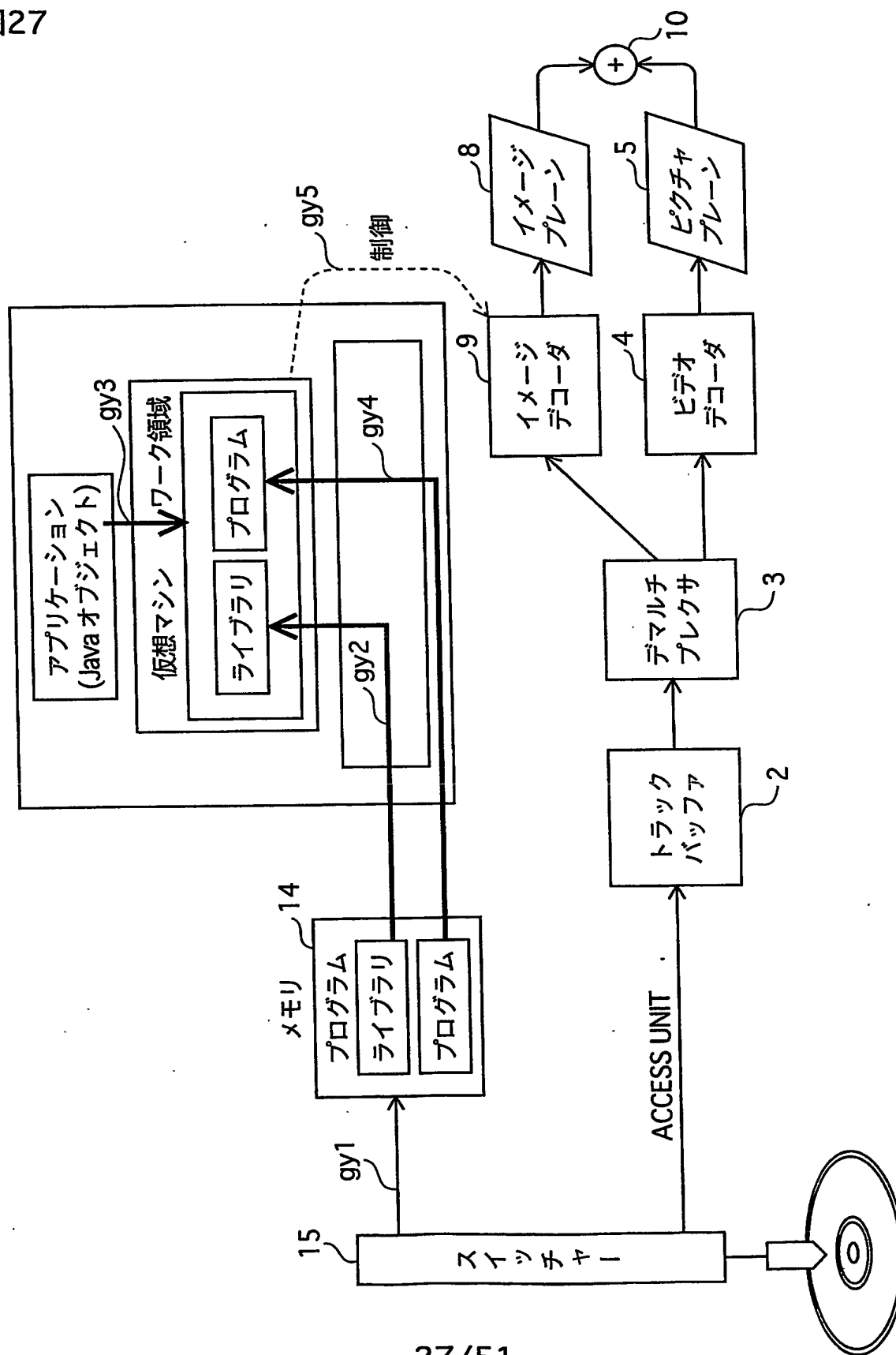


図28

Java 言語が対象とする制御ソフトウェアのレイアモデル

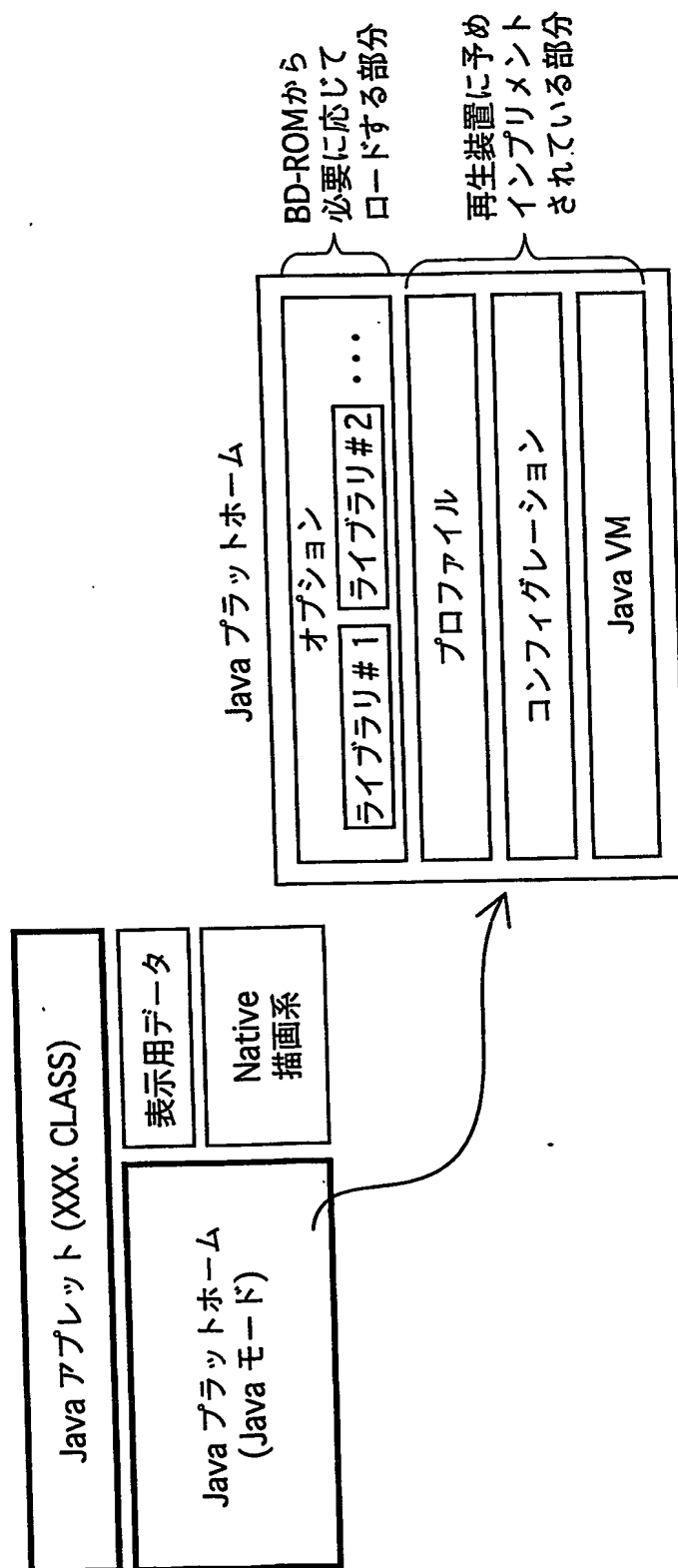


図29

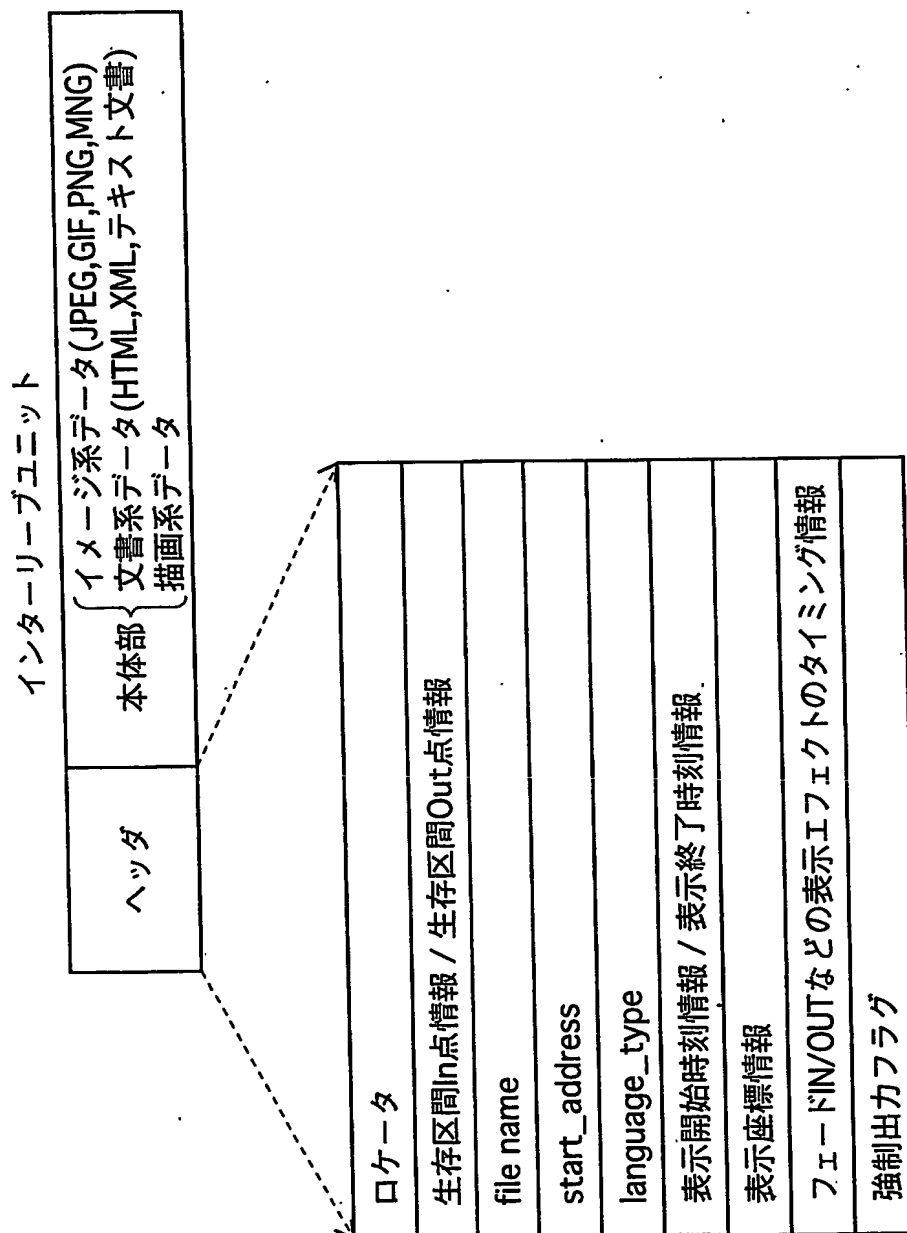


図30

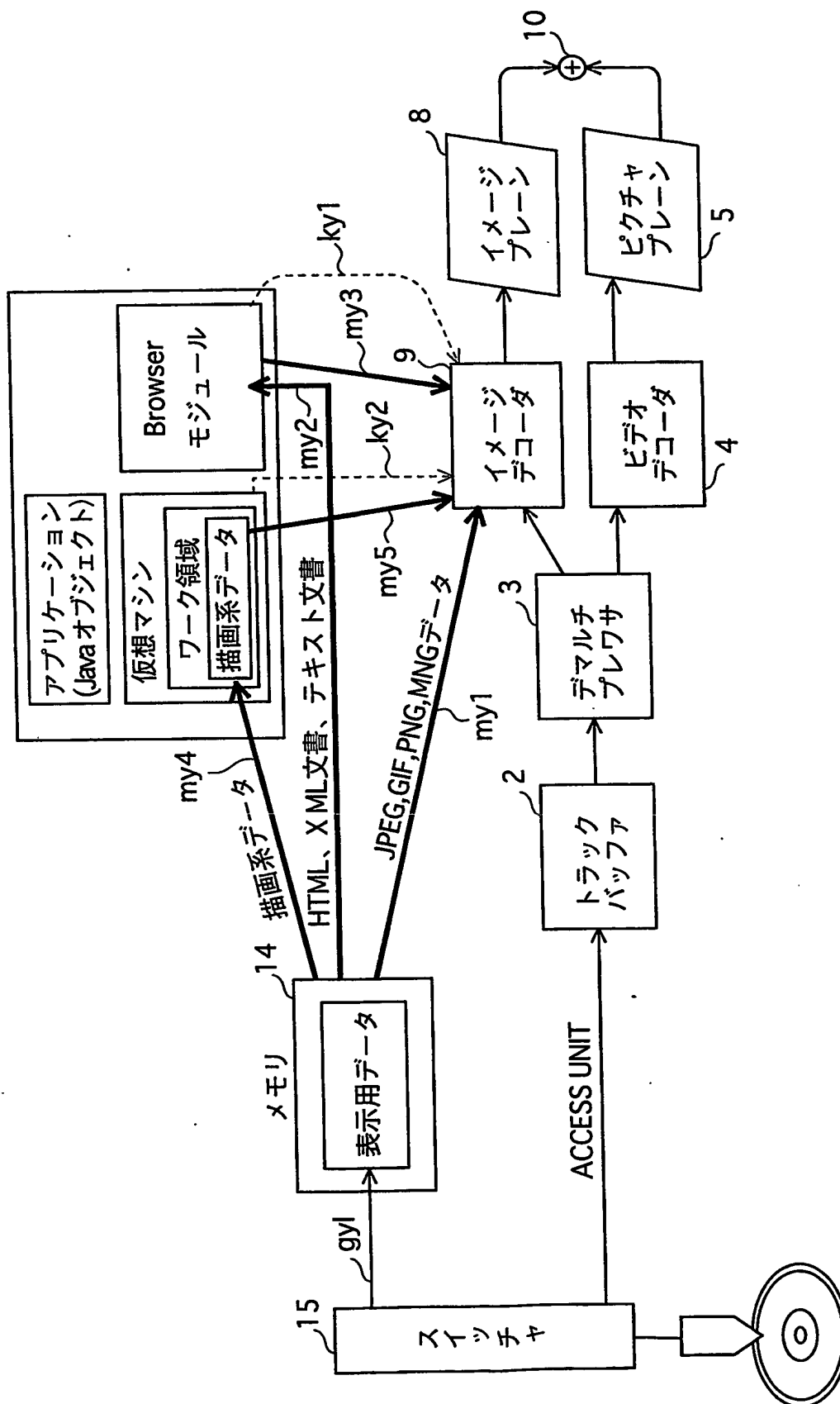


図31

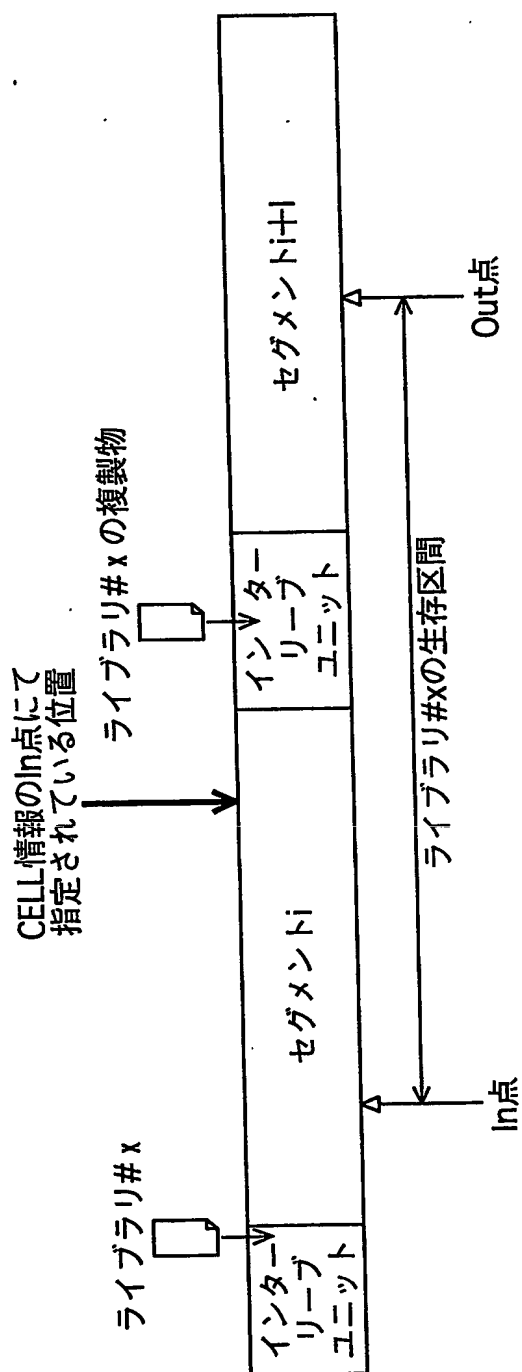


図32

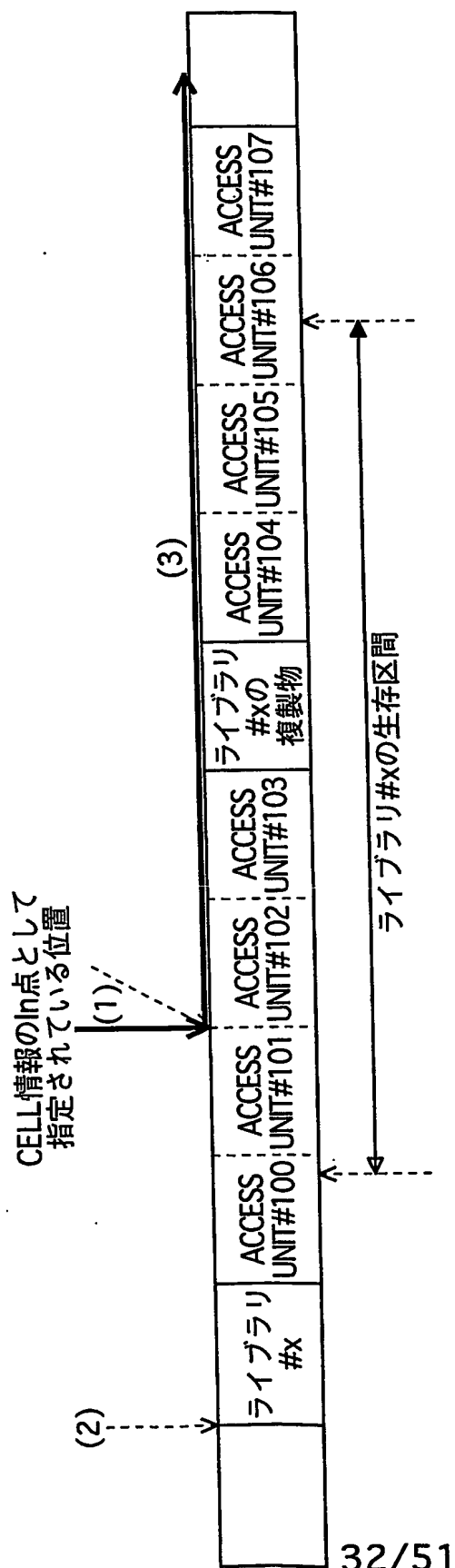


図33

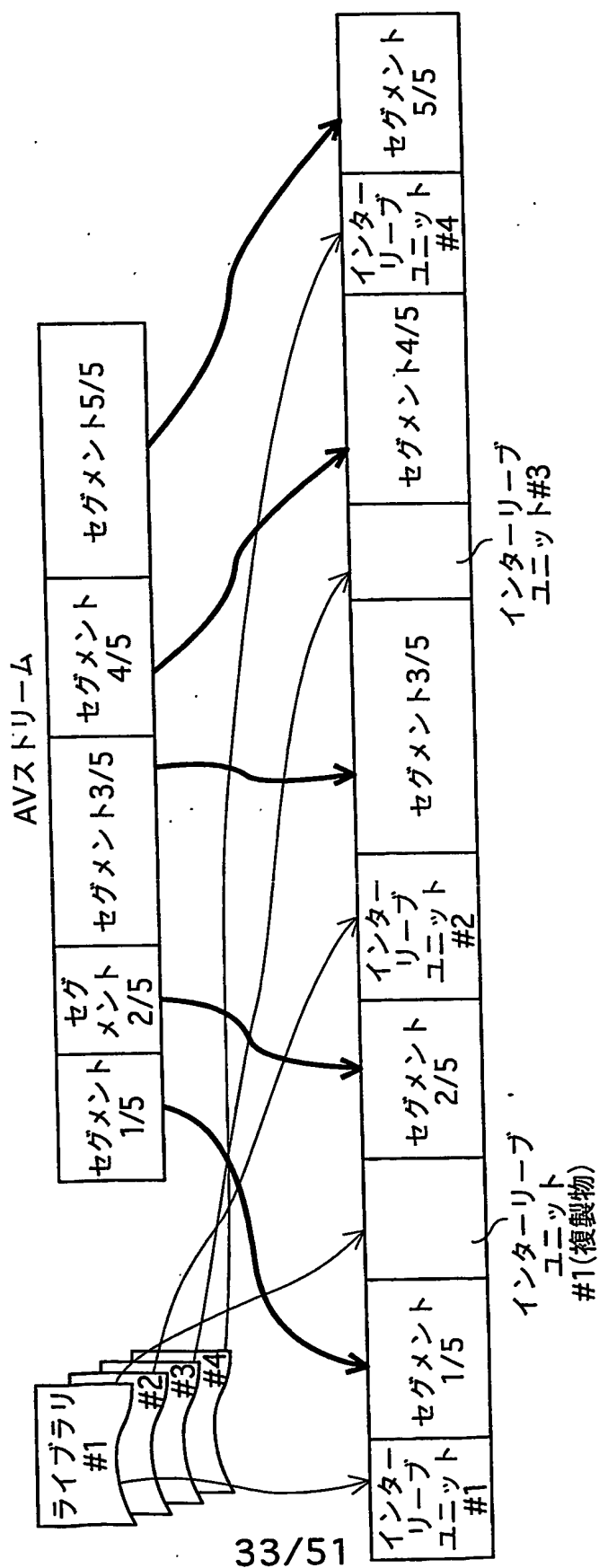




図34

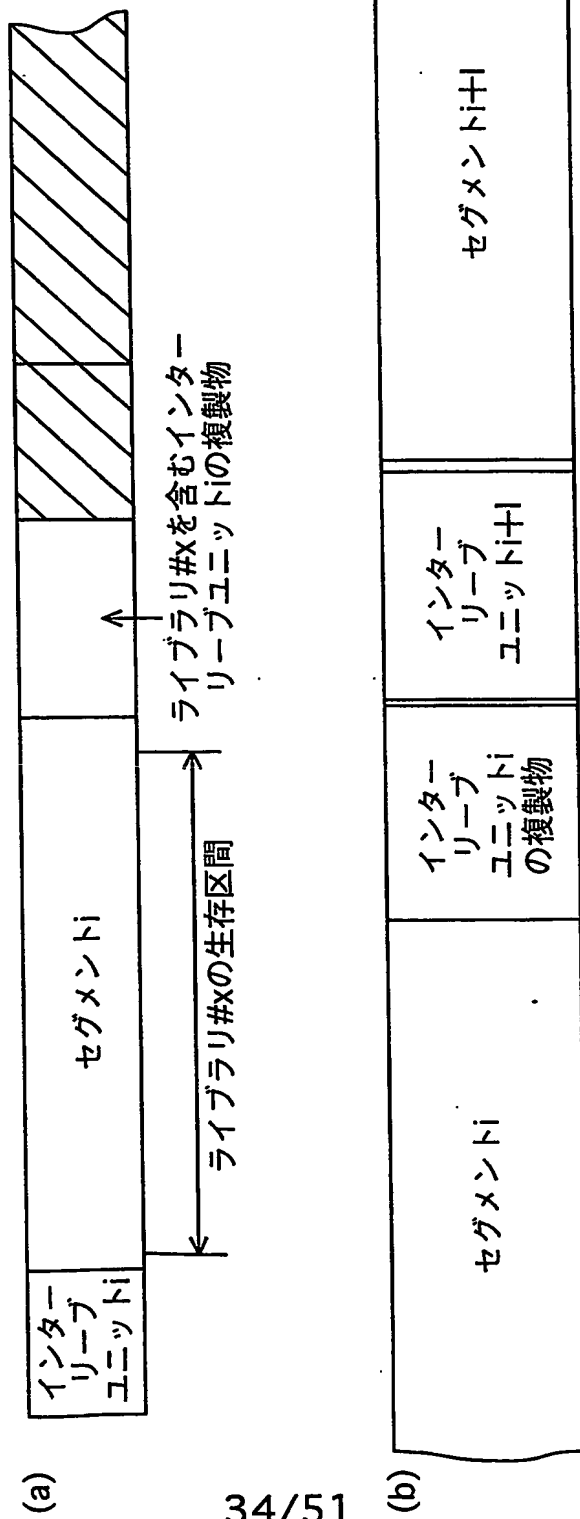


図35

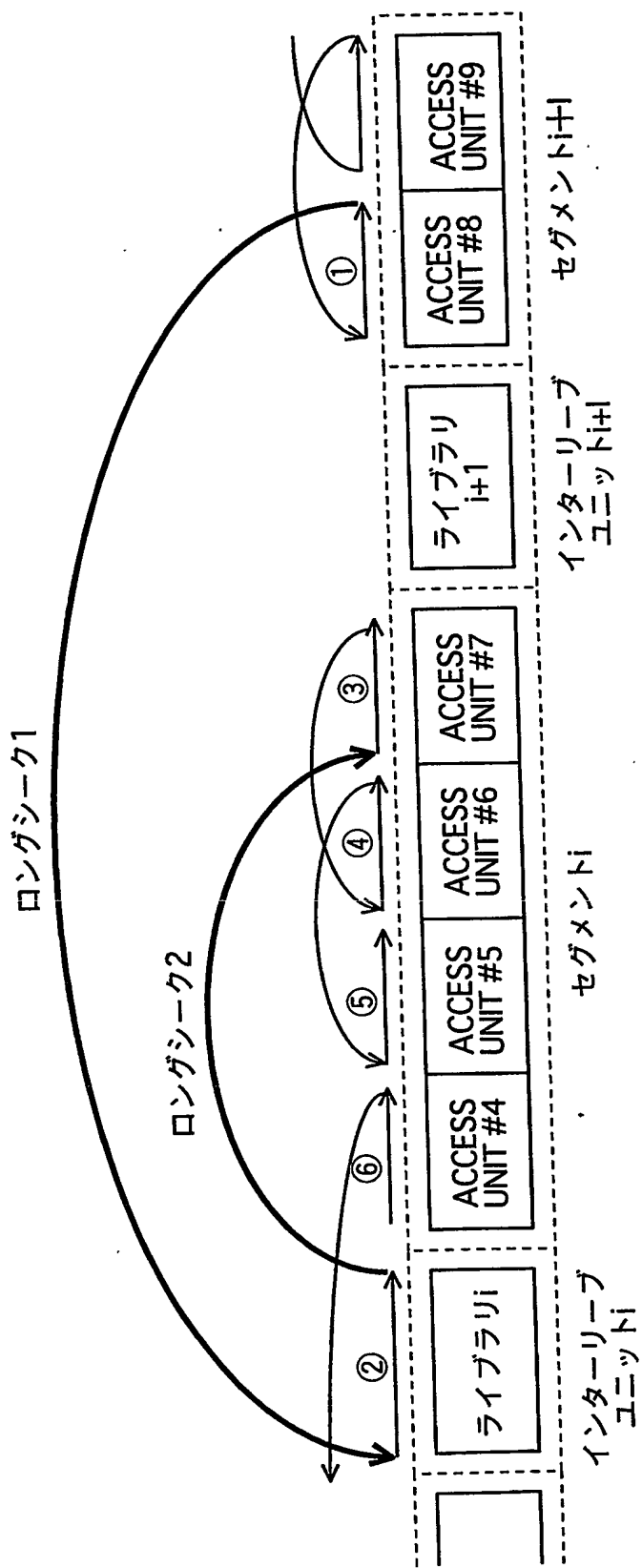


図36

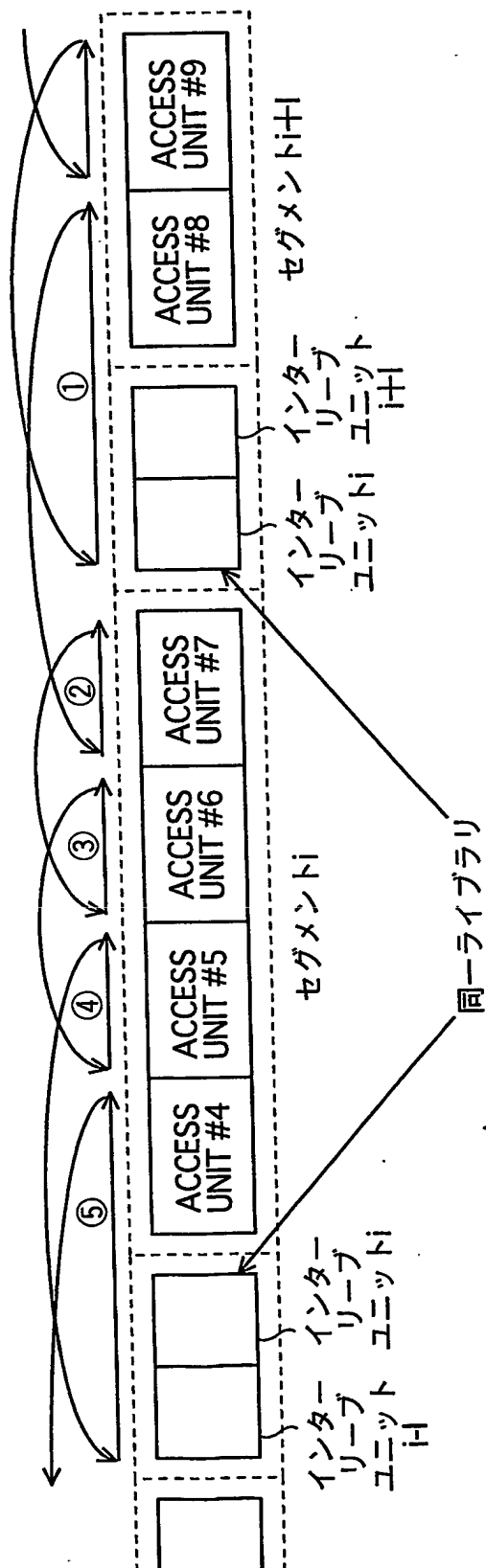


図37

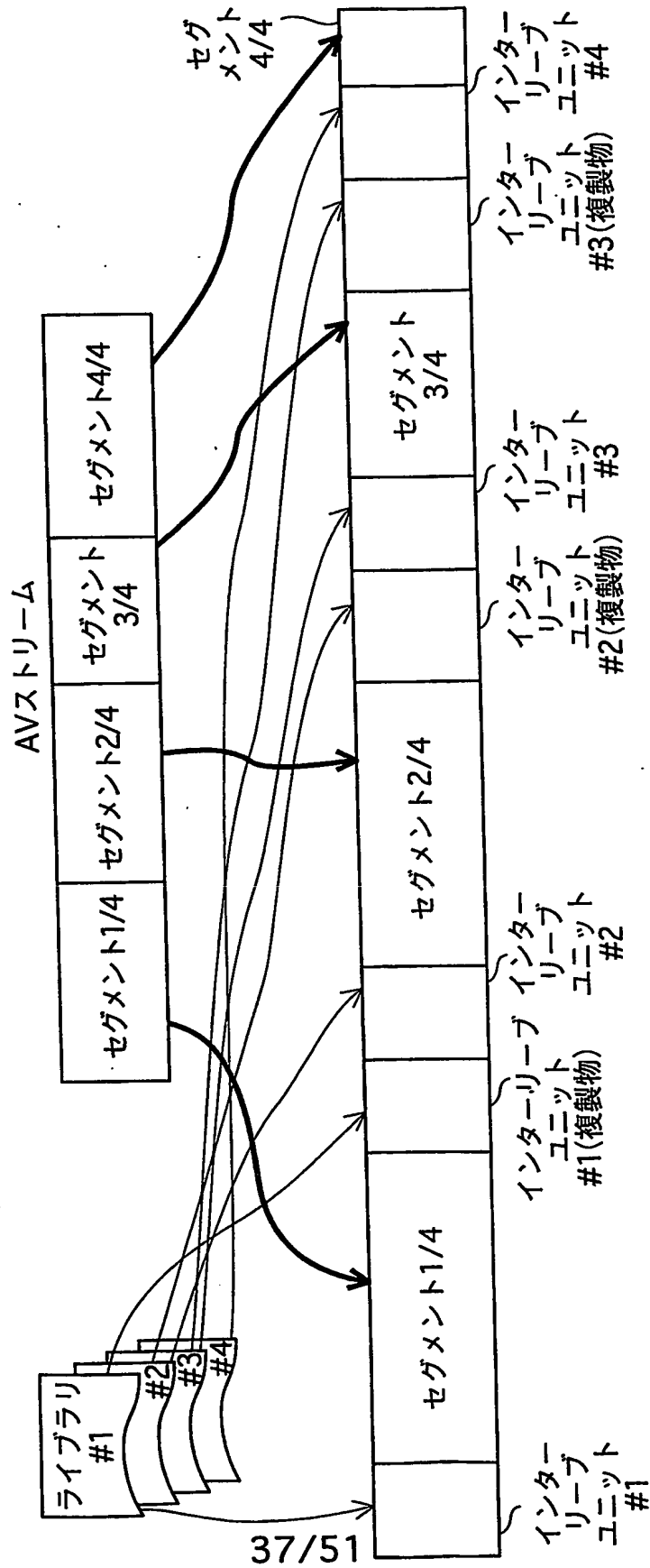


図38

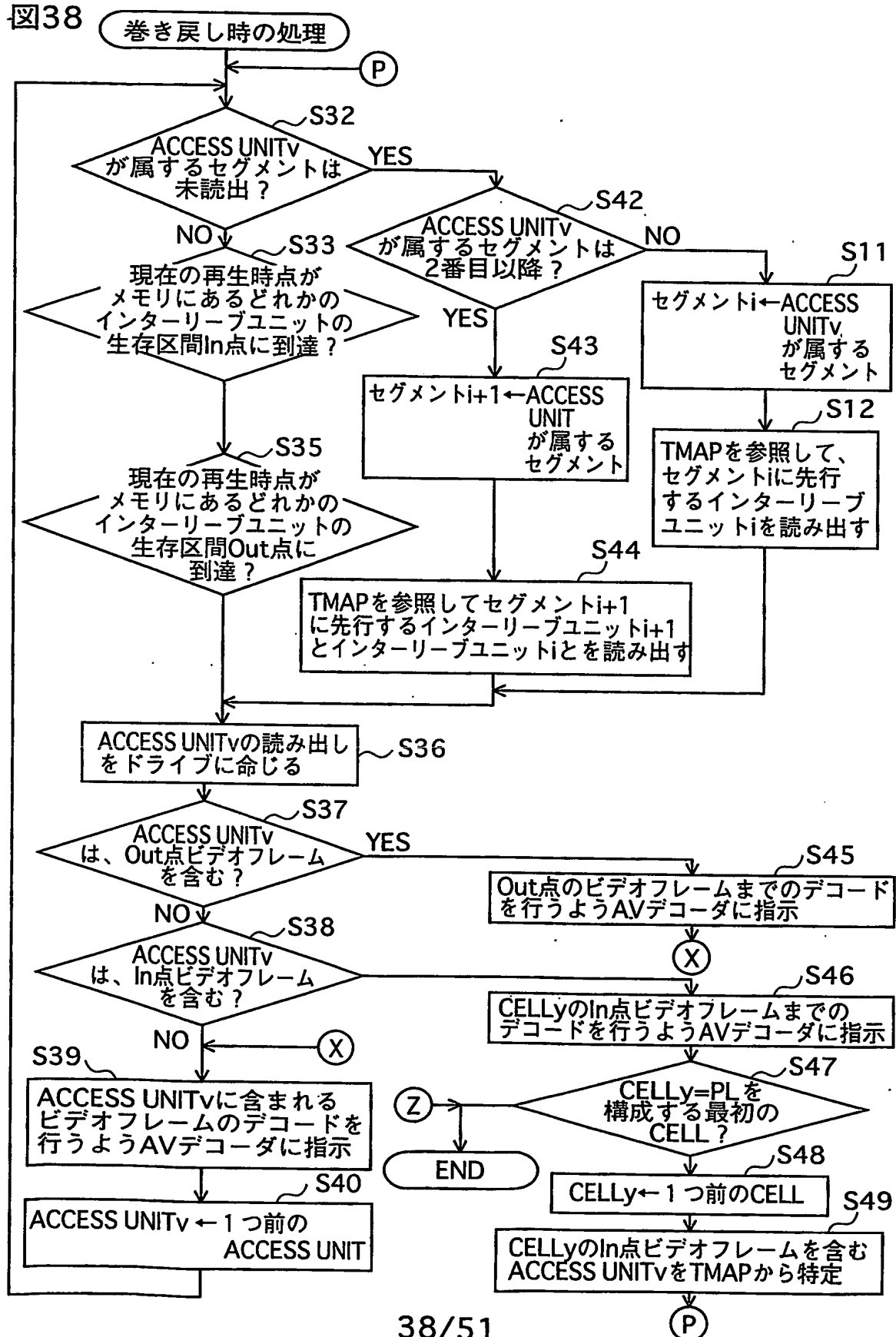
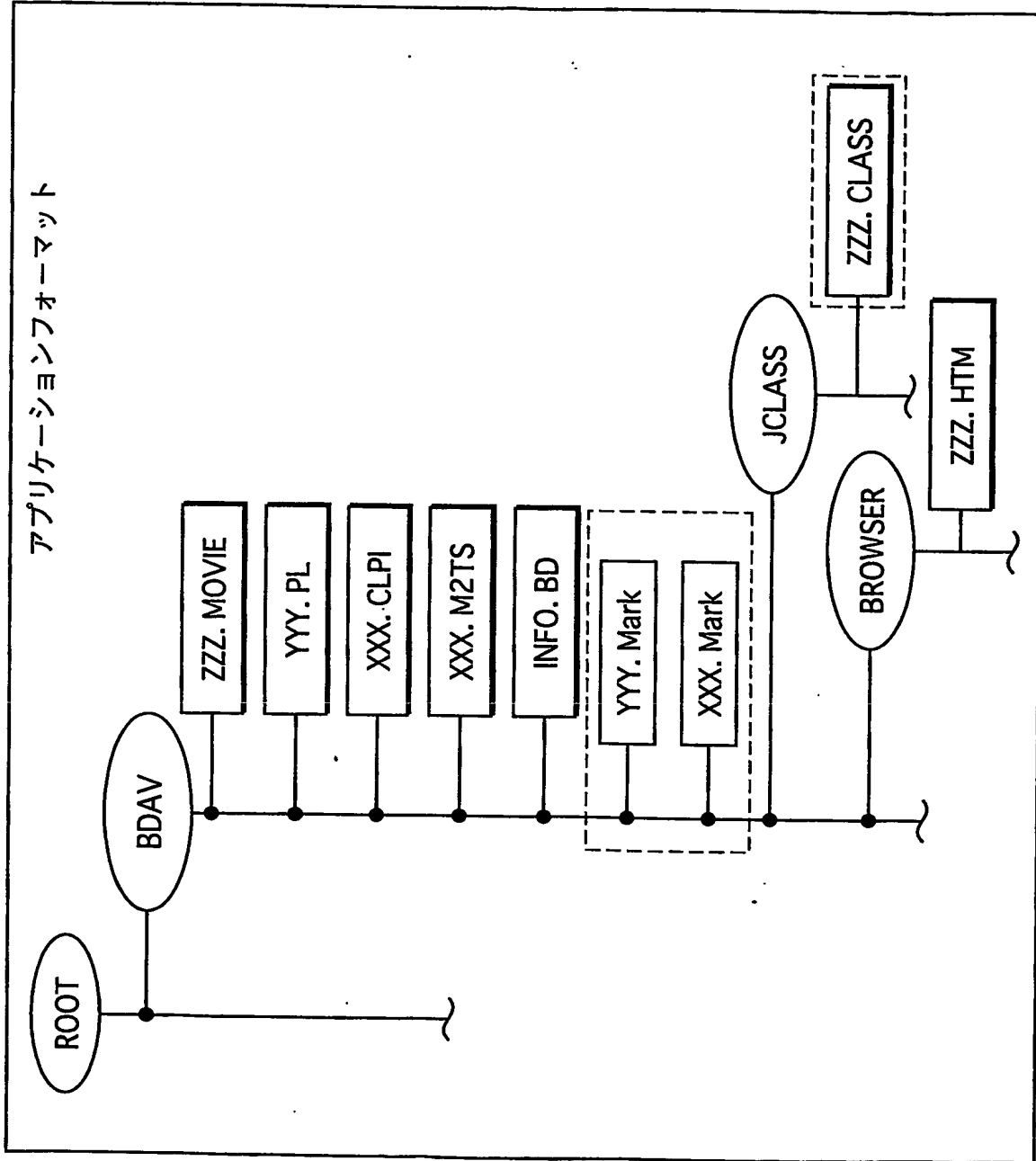


図39

アプリケーションフォーマット



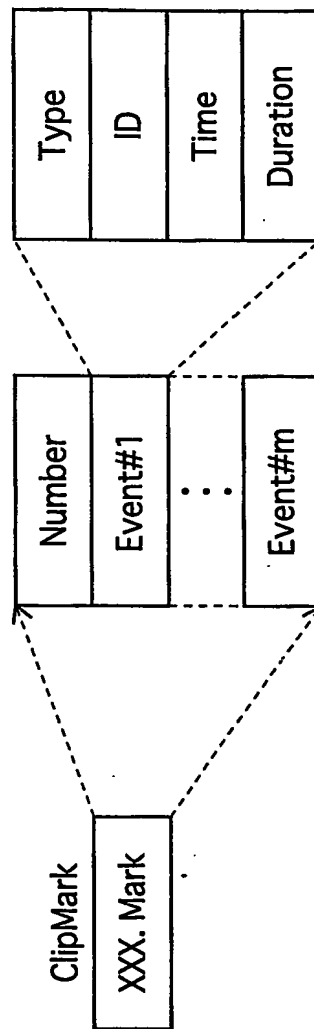
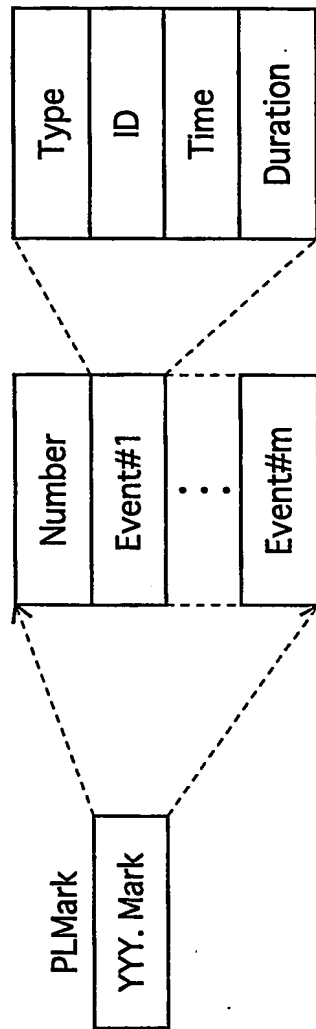
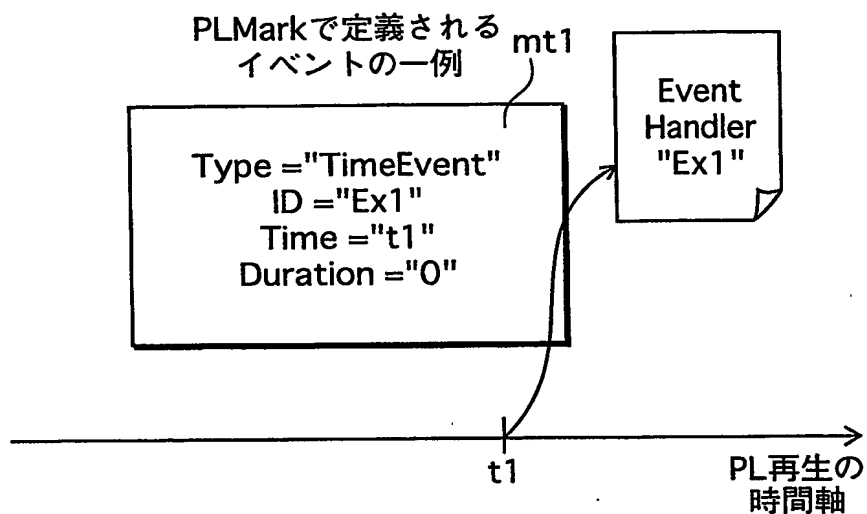


図41

(a)



(b)

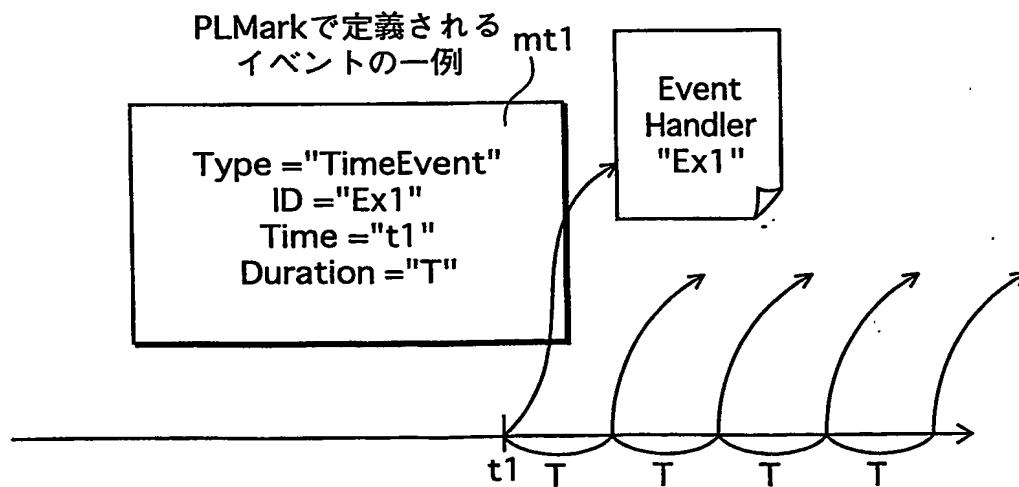




図42

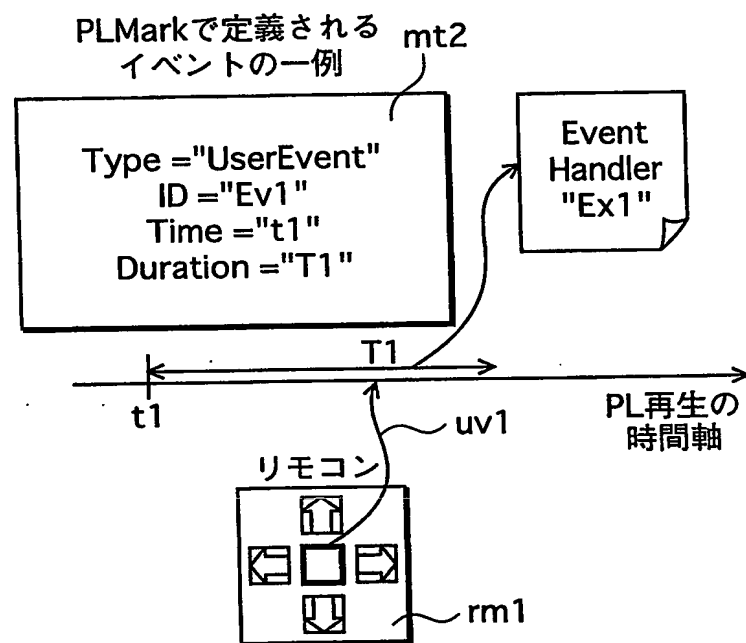


図43

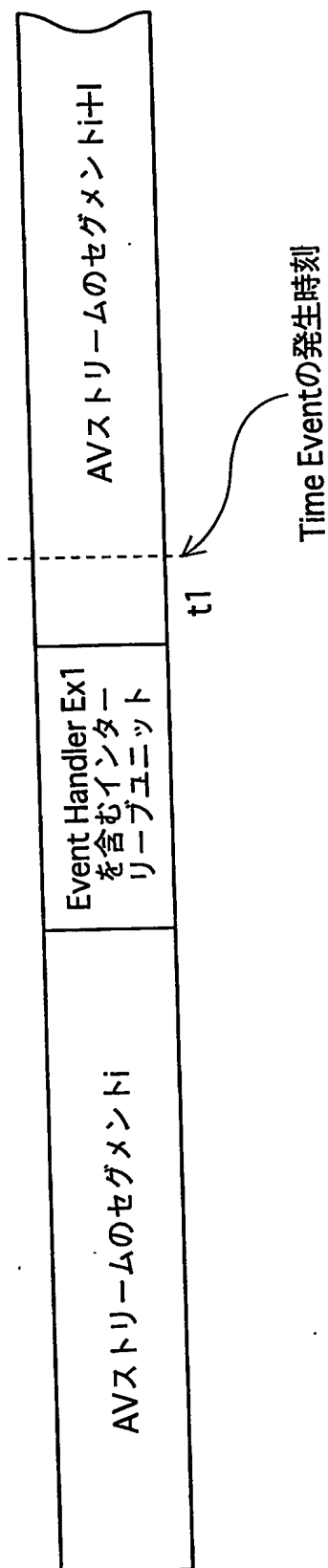


図44

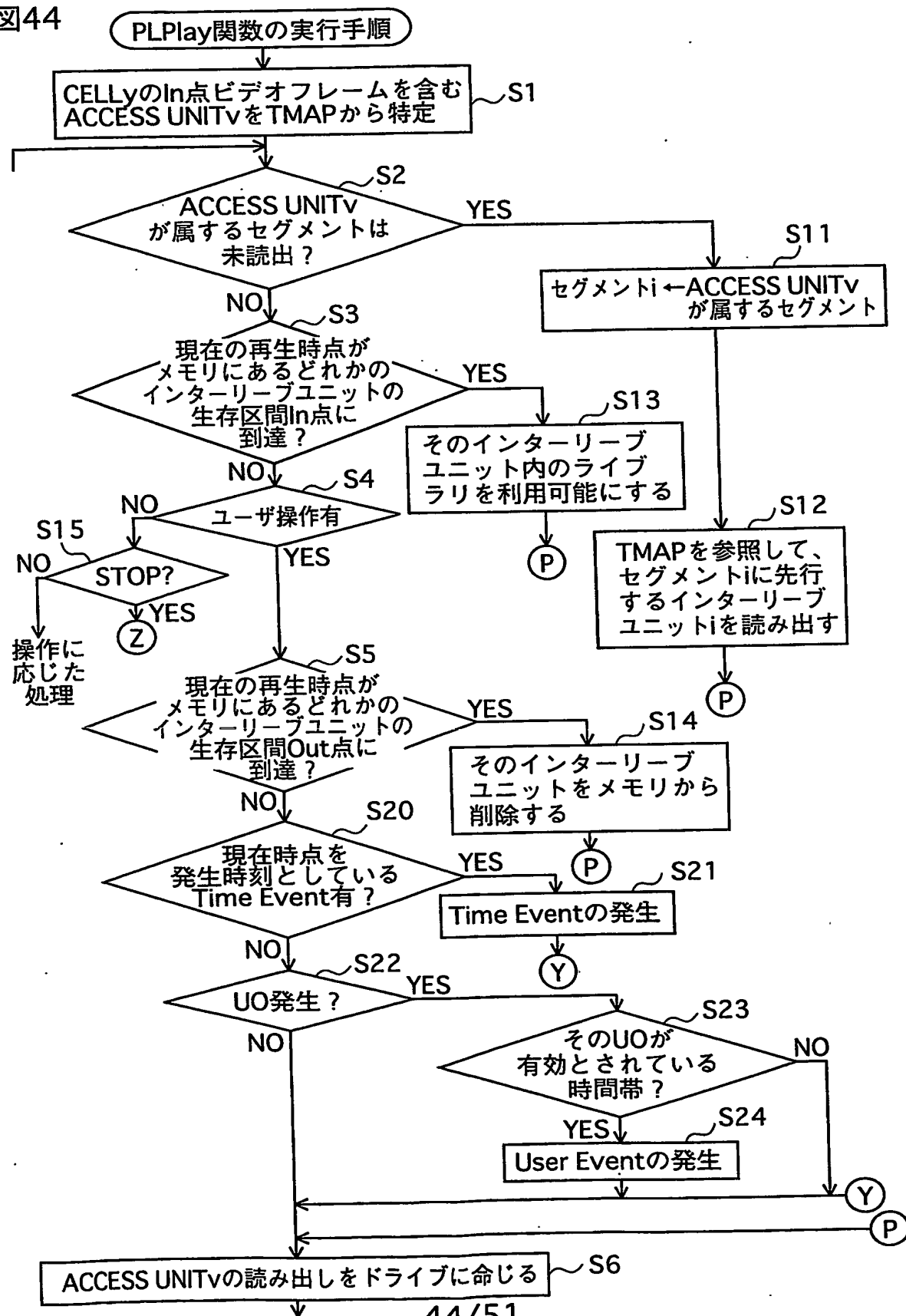


図45

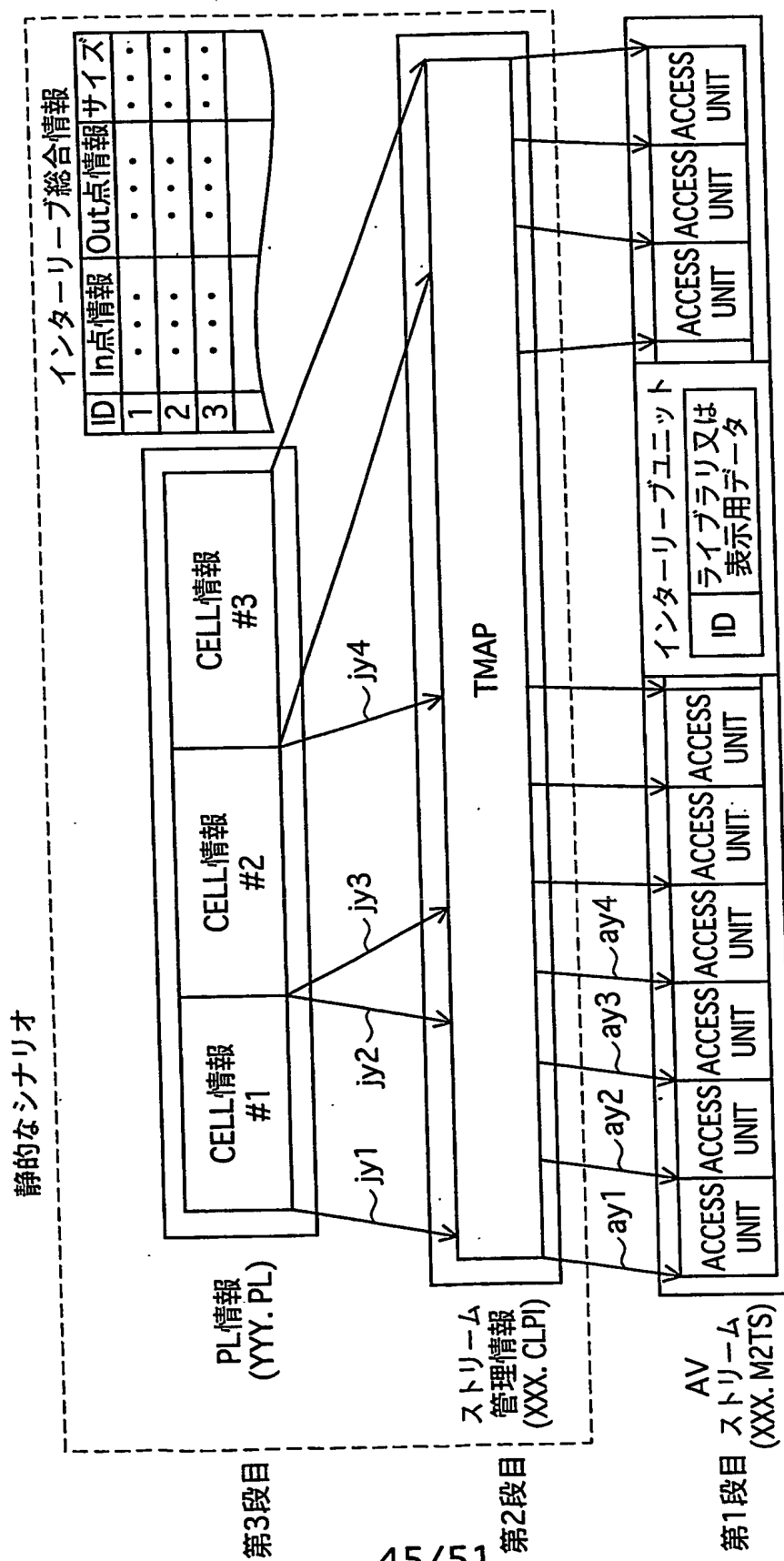


図46

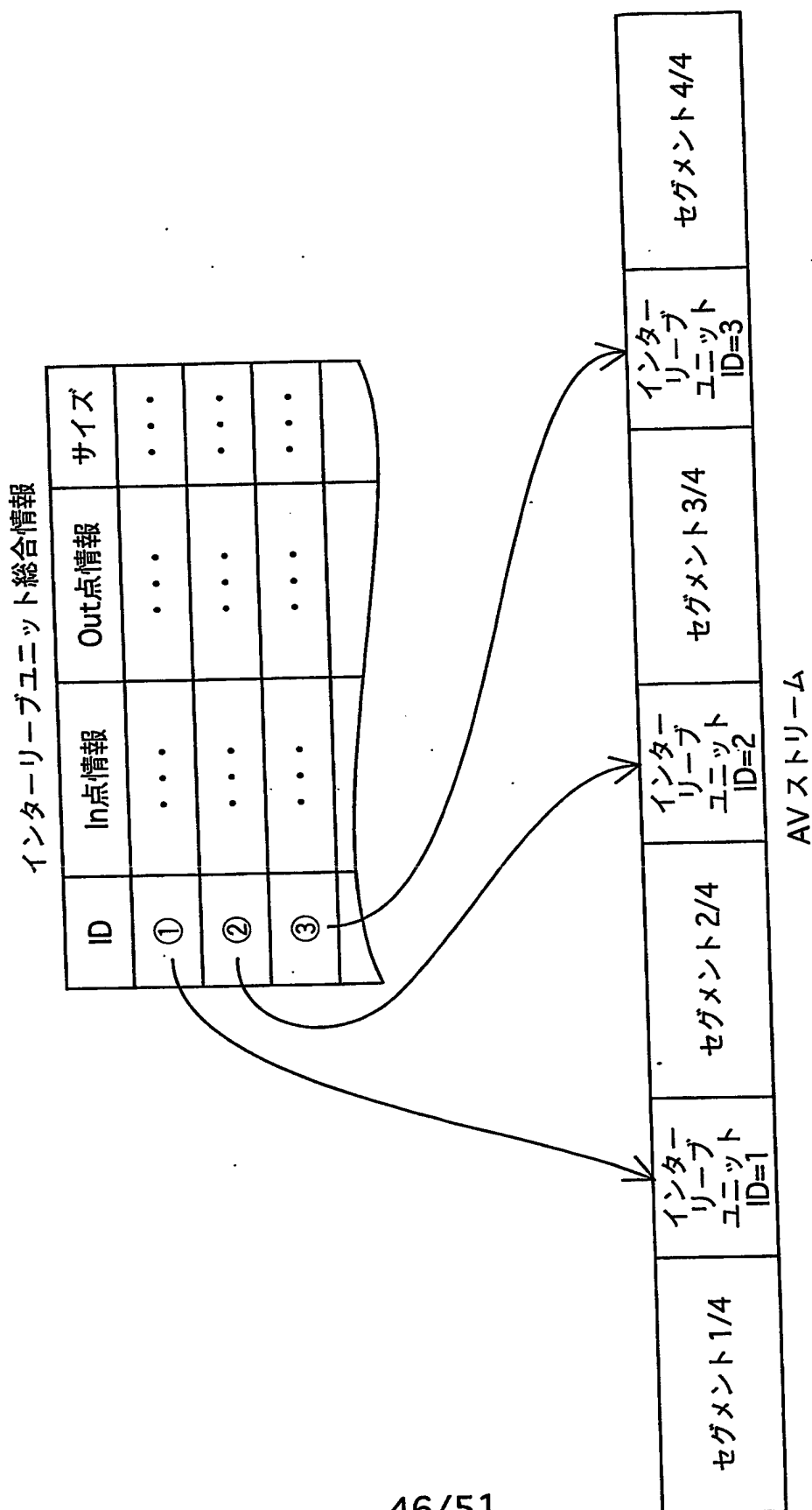


図47

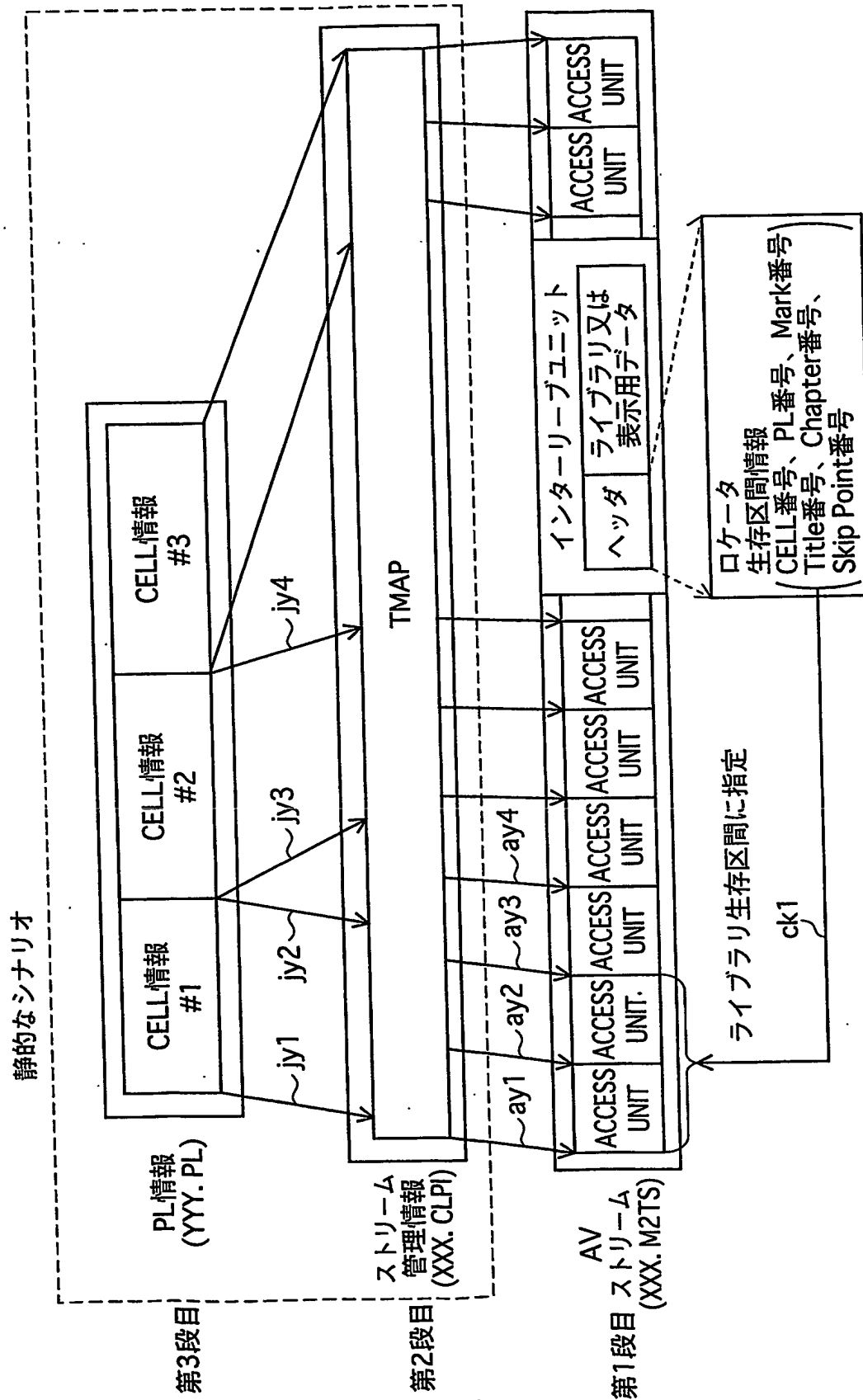


図48

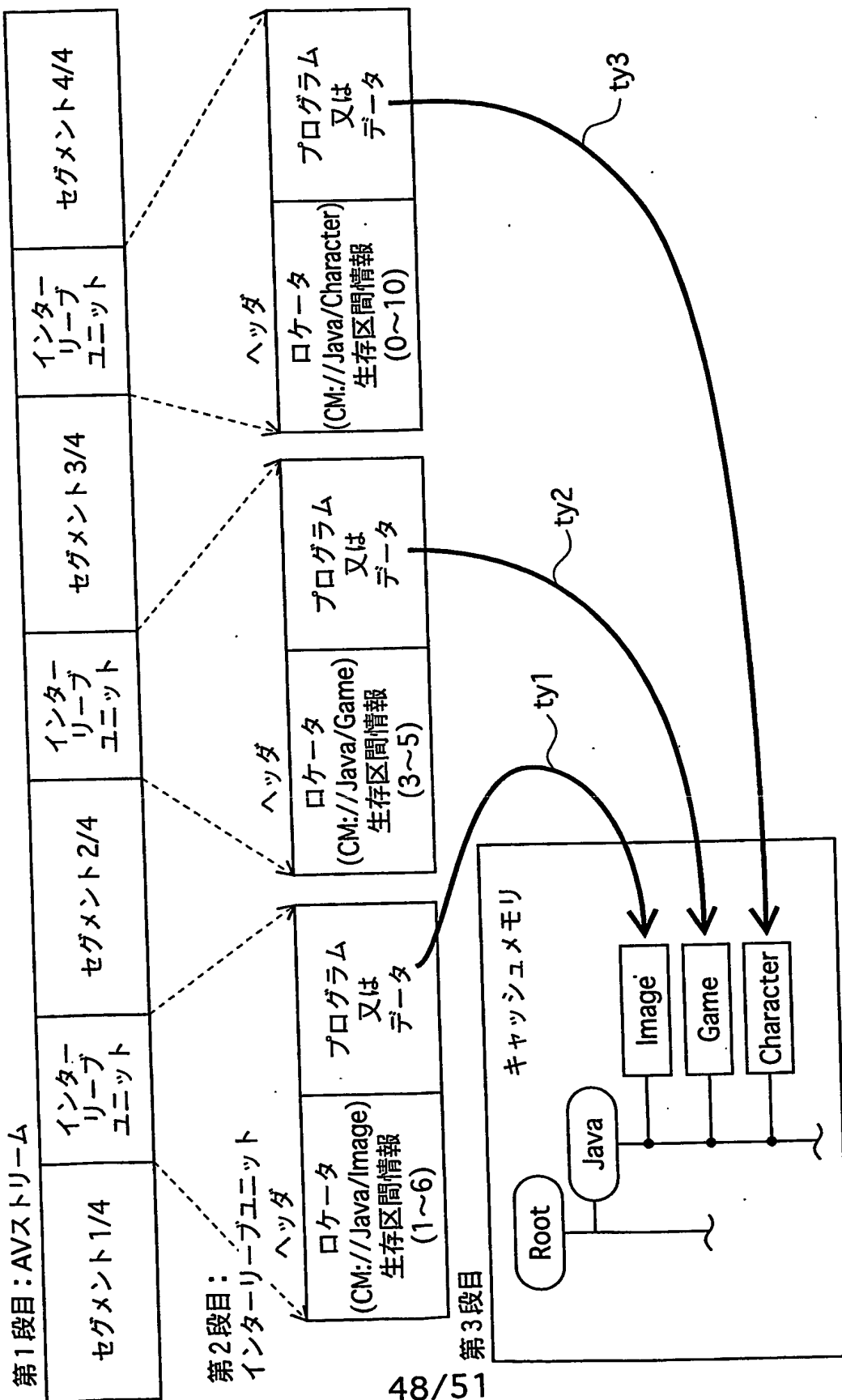


図49

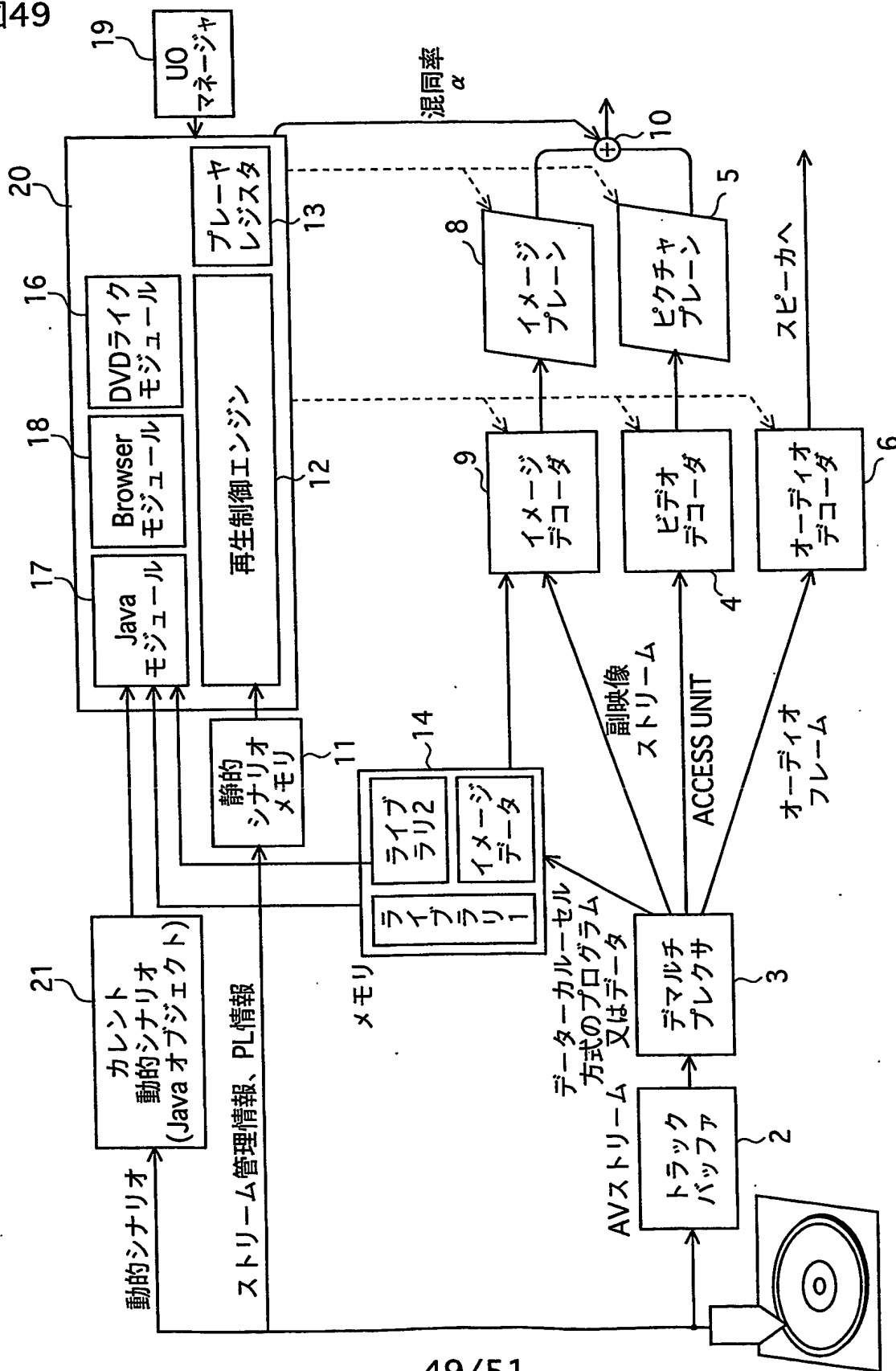




図50

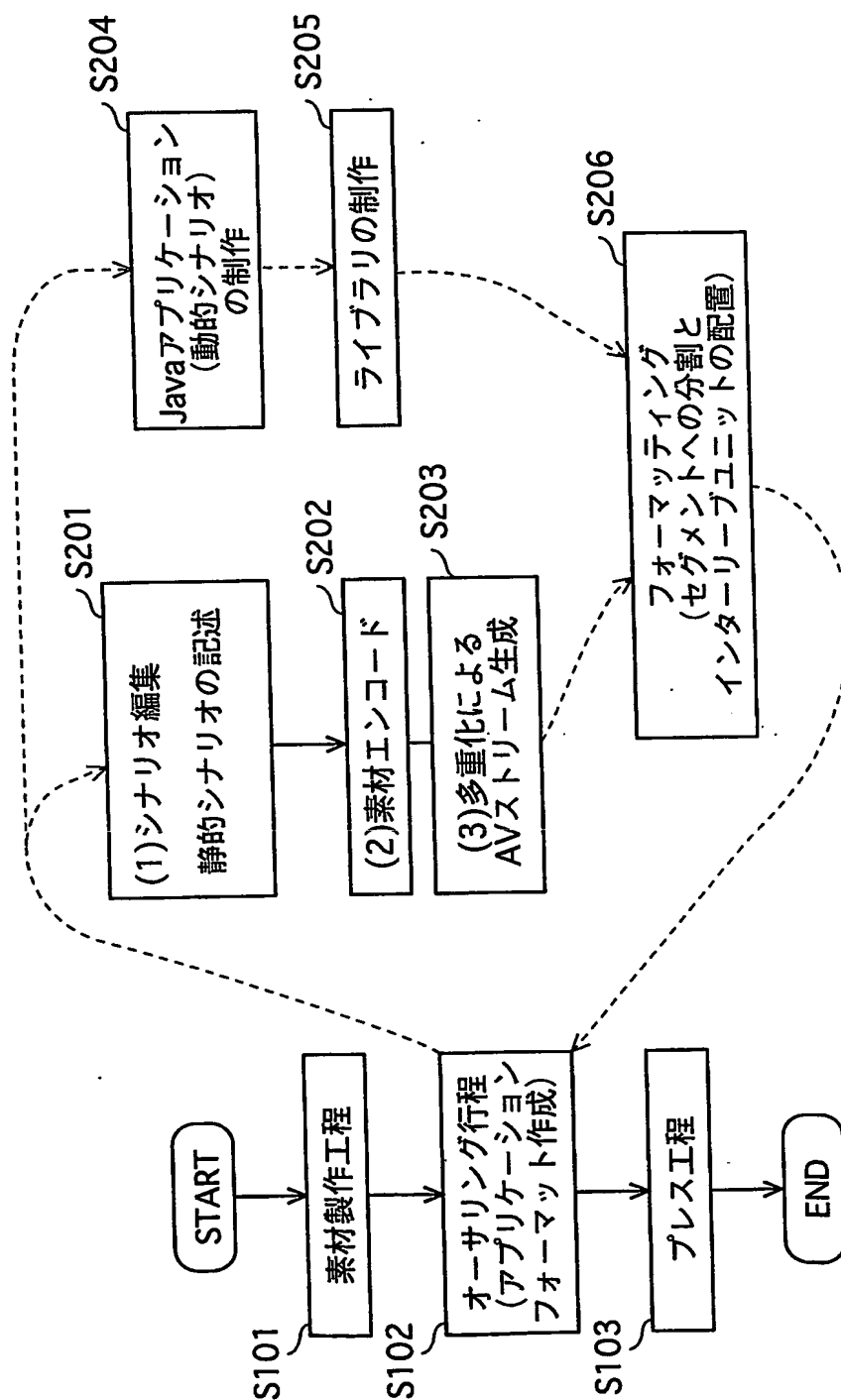


図51

## インターリーブユニット

ヘッダ	イメージ系 データ ( JPEG, GIF ) ( PNG, MNG )	文書系 データ ( HTML, XML ) ( テキスト文書 )	描画系 データ
-----	---	---	------------

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP03/13026

## A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl<sup>7</sup> G11B20/10, 27/00, G06F3/06, H04N5/92

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl<sup>7</sup> G11B20/10, 27/00, G06F3/06, H04N5/92

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho 1922-1996 Toroku Jitsuyo Shinan Koho 1994-2003

Kokai Jitsuyo Shinan Koho 1971-2003 Jitsuyo Shinan Toroku Koho 1996-2003

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2000-137632 A (Square Co., Ltd.), 16 May, 2000 (16.05.00), All pages; all drawings & US 6604181 B1	1-37
A	JP 2002-084501 A (LG Electronics Inc.), 22 March, 2002 (22.03.02), Par. Nos. [0005], [0023]; Fig. 6 & US 2001/0056580 A1	1-37
A	JP 2000-293972 A (Pioneer Electronic Corp.), 20 October, 2000 (20.10.00), Par. Nos. [0025] to [0030], [0053] to [0056]; Figs. 1, 4 & EP 897178 A2	1-37

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document but published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search  
04 December, 2003 (04.12.03)

Date of mailing of the international search report  
16 December, 2003 (16.12.03)

Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

## A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl<sup>7</sup> G11B20/10, 27/00, G06F 3/06, H04N 5/92

## B. 調査を行った分野

## 調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl<sup>7</sup> G11B20/10, 27/00, G06F 3/06, H04N 5/92

## 最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1922-1996年  
 日本国公開実用新案公報 1971-2003年  
 日本国登録実用新案公報 1994-2003年  
 日本国実用新案登録公報 1996-2003年

## 国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

## C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	JP 2000-137632 A (株式会社スクウェア) 2000. 05. 16 全頁、全図 & US 6604181 B1	1-37
A	JP 2002-084501 A (エルジー電子株式会社) 2002. 03. 22 段落【0005】、【0023】、第6図 & US 2001/0056580 A1	1-37

☒ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

## \* 引用文献のカテゴリー

「A」 特に関連のある文献ではなく、一般的技術水準を示すもの  
 「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの  
 「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)  
 「O」 口頭による開示、使用、展示等に関する文献  
 「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

## の日の後に公表された文献

「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの  
 「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの  
 「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの  
 「&」 同一パテントファミリー文献

## 国際調査を完了した日

04. 12. 03

## 国際調査報告の発送日

16.12.03

## 国際調査機関の名称及びあて先

日本国特許庁 (ISA/J P)  
 郵便番号100-8915  
 東京都千代田区霞が関三丁目4番3号

## 特許庁審査官 (権限のある職員)

齋藤 哲

5 Q

4 2 3 2

電話番号 03-3581-1101 内線 3590

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	JP 2000-293972 A (パイオニア株式会社) 2000.10.20 段落【0025】-【0030】, 段落 【0053】-【0056】、 第1図, 第4図 & EP 897178 A2	1-37